Selma Čeliković, BSc

Control of Pharmaceutical Production Plants

Master's Thesis

to achieve the university degree of

Dipl.-Ing.

Master's degree programme: Electrical Engineering

submitted to

Graz University of Technology

Supervisor

Univ.-Prof. Dipl-Ing. Dr.techn. Martin Horn

Institute of Automation and Control
Head: Univ.-Prof. Dipl-Ing. Dr.techn. Martin Horn

Graz, October 2020

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date	Signature

Abstract

This thesis outlines a development, implementation and application of diverse control concepts for a continuous tablet manufacturing plant. The considered plant consists of the hot-melt-extrusion line, involving feeding, extrusion, cooling, pelletisation and transport, and the direct-compaction line, involving feeding, blending and tablet compaction. Proper continuous plant operation is ensured when the following control objectives are fulfilled. First control objective is to keep the API concentration in the allowed range continuously along the plant. This is ensured via application of the concentration controller combining Smith predictor structure and PID controller with anti-windup. Material exhibiting the properties outside the allowed API concentration range is discharged by means of the hysteresis controller. Second control objective is a production of tablets with technical properties within the pre-specified, valid range. TP MPC copes with this objective, tending to keep TP fill level constant by adjusting the TP turret speed and the blender inlet mass flow. Third control objective is an uniform and reliable transport of material from the HME to the DC line. Mass flow controller is utilized to prevent the undesired process events, such as exhaustion or exceeding of material between the two lines. DC line control concept, combining the TP MPC, concentration and discharge control, is developed and examined both in the simulation and on the real system. For the development of the complete plant control concept, three candidates for the master unit are suggested: extruder, TP and no master unit. Depending on the active master unit manipulated variables of the individual controllers change, while the control objectives stay the same. Functionality of the complete plant control concept is examined via same test scenario in the simulation for each proposed master unit.

Acknowledgment
This master thesis was a part of PharmControl project partially funded by Austrian Research Promotion Agency FFG, program “Produktion der Zukunft”, project number 858704.

Contents

List of Figures

1 Introduction

During the last years, pharmaceutical companies have put a focus on a continuous manufacturing of the pharmaceutical products. Traditionally, these were produced via batch based processes.
Fig. 1.1 depicts differences between these two production approaches: A batch based production is characterized by the time-outs between the individual process steps. Quality tests of the intermediate/final product take place during these time-outs. If the quality of test product meets predefined expectations, it can be proceeded to the next step. Otherwise, it needs to be discharged or reprocessed.
On the other hand, a continuous production implies an uninterrupted process, from the raw material to the final product. Thereby, a quality of the intermediate/final product in between the individual process steps needs to be tracked during the plant operation.
As a consequence of the time-outs between the process steps, batch processes exhibit low production speed. US Food and Drug Administration (FDA) researches estimate, that a switch to the continuous production would reduce the production duration from months to days. In that sense, this switch would potentially lead to the higher efficiency, higher production speed and decreased production costs.
According to [xtalks, 2016], a time duration between the individual process steps seems to be directly proportional to the risk of the potential contamination or human error, which negatively affects the reliability of batch production.
[Lee et al., 2015] suggests that a switch from the batch to the continuous production has a potential to increase agility, flexibility and robustness in the pharmaceutical production. Additionally, [Sacher et al., 2019] recognizes a switch to the continuous production as a natural extension, due to the fact the most common pharmaceutical unit operations are continuous by its design.

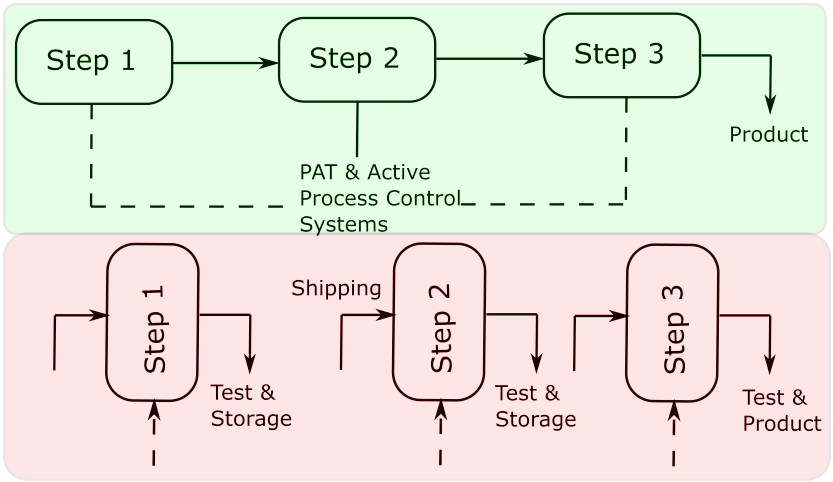


Figure 1.1: Comparison of batch and continuous production: Simplified concept of continuous (green) and batch (pink) manufacturing process [Lee et al., 2015].

[Lee et al., 2015] proposes the following concept of a continuously operated production plant: Individual unit operations should be joined to an integrated process. A real-time data provided by the process analytical technology (PAT) tools should be used to monitor process performance and detect the potentially dangerous process events. In order to avoid a quality degradation of the final product, caused by the raw material or the process variability, process control should be implemented during the plant operation. [Rehrl et al., 2016] suggests, that the development of control strategies is an imperative for maintaining the appropriate continuous plant operation. Also, [Su et al., 2018] indicates that the robust and reliable controllers are crucial for providing the quality control and risk managements within a continuous plant.
In the continuous production regime, product flows directly from one to another process step. Therefore, it is important to ensure that the intermediate product meets prescribed quality attributes within each step. Any out-of-specification (OOS) material needs to be detected and discharged immediately. [Rehrl et al., 2018] proposes an approach based on the residence time distribution (RTD) model for the detection of OOS material. This approach was investigated and utilized within this work.
Up to this point, several control techniques for continuous plants have been designed, developed and implemented. Yet, a majority of proposed techniques involves solely well known, standard control techniques. Therefore, it is compelling to investigate the application of advanced, and from a control engineering perspective more interesting control techniques, such as model predictive control (MPC). An MPC application offers several advantages, like a straightforward consideration of the constraints, as well as the simple expandability to the multi-input multi-output (MIMO) systems. Furthermore, MPC is particularly suitable for controlling the systems of high complexity.
In that sense, [Rehrl et al., 2016] develops and compares standard and advanced control techniques for the continuously operated feeding-blending-unit (FBU) in the simulation. [Celikovic et al., 2019] continues this work by developing and investigating the proposed control concepts on the real system. They report that MPC shows advantageous behavior compared to the standard proportional integral (PI) and model-based feed forward control techniques. Due to the incorporated optimization problem which is solved in each iteration, MPC requires higher computational effort in comparison to the standard control techniques. Yet, [Bhaskar et al., 2017] claims, that despite its complexity it is reasonable to implement an MPC, as its application leads to the significantly better closed-loop performance.

This master thesis outlines the development of an automated continuous tablet manufacturing plant, with a focus on the design, implementation and real-system application of an adequate control concept. The examined production plant consists of two lines: A hot-melt-extrusion (HME) line and a direct-compaction (DC) line. Pellets based on the active product ingredient (API) and polymer matrix are produced and quality proofed within the HME line. The produced pellets are then supplied to the DC line. Within this line, pellets are blended with a pre-mix material, quality proofed, and finally compressed to the tablets. Fig. 2.1 provides a detailed schematic overview of the continuous tablet manufacturing plant, including the individual process units and the utilized PAT equipment.
Process modeling represents a starting point for the development of the control concepts for the investigated continuous plant. For the modeling purposes, a complete plant is decomposed into the individual modeling units, which correspond to the individual unit operations, i.e. feeding, extrusion and cooling & pelletisation in the HME-, and feeding, blending and TP in the DC line. Data-driven model identification is performed by means of the MATLAB System Identification Toolbox App [SysIdTool, 2020] and MATLAB function *ssest* [ssest, 2020]. The data required for the identification procedure is collected via laboratory experiments executed on the real system. The identified models are compared to the real system. When a satisfactory conformity between the real-(measured) and model-estimated behavior is obtained, identified models can be utilized for the development of the control concepts.
In order to ensure the appropriate continuous plant operation, several control objectives need to be fulfilled. First control objective is to keep the product API content within the allowed range. This control objective is addressed via application of concentration controller. As the identified transfer functions relating inlet- and outlet concentration, both in HME and DC line, exhibit a certain time delay, concentration controller is realized as a combination of Smith predictor structure involving a standard PID controller. In case of disturbances, which cannot be attenuated by the application of concentration controller, material with the API content outside the allowed range needs to be discharged. Discharge is realized by means of the hysteresis controller.
Second control objective is a production of tablets with constant and satisfactory properties. Experiments have shown, that if the TP hopper fill level sinks below a certain value, a quality of the produced tablets is seriously affected. On the other hand, it is also observed, that a change of TP turret speed can influence the fill level, without negatively affecting the tablet properties. Therefore, an MPC for the fill level control is designed and implemented. MPC manipulates blender inlet mass flow and TP turret speed in order to keep the TP fill level close to its reference.
Initially, a control concept is developed solely for the DC line. Blender and TP model are concatenated to a control-oriented state-space-representation and this representation is utilized for the TP MPC application. DC line control concept involving the concentration, discharge and fill level controller, is designed, simulated and examined on the real system.
Finally, DC line control concept is extended to a control concept for the complete continuous plant. This leads to an additional control objective, i.e. uniform and reliable pellet transport between the two lines. This objective is targeted by the application of a mass flow controller. Additionally, concentration and discharge controller are implemented in the HME line. Three different concepts (A, B and C)

for the control of the complete line are suggested. These concepts distinguish in the active master unit, i.e. extruder, TP or no master unit, respectively. Finally, a control action of each control concept is examined in the simulation by an artificially introduced feeder disturbance scenario. Obtained results are compared in regard to the important process quantities and analyzed in view of the further work on the complete continuous plant.
The remainder of the thesis is structured as follows:
Chapter 2 provides an overview of the continuous plant structure, process flow and the utilized PAT equipment. Details of continuous plant modeling can be found in the Chapter 3. A theoretical background of the employed control concepts, development of the individual controllers, and application of DC line control concept in the simulation and on the real system are introduced in the Chapter 4. In the Chapter 5, different control concepts for the complete continuous plant are proposed, designed and examined in the simulation. Finally, Chapter 6 concludes the thesis, provides a summary and outlines the potential next steps.

2 Continuous Tablet Manufacturing Plant: Technical Description

This Chapter provides a technical description of the continuous tablet manufacturing plant. The investigated plant consists of two lines, HME and DC line. Both lines involve several unit operations, namely, feeding, extrusion, cooling, pelletisation, discharge and pneumatic conveying in the HME line, and feeding, blending, discharge and tablet compression within the DC line. Section 2.1 provides a detailed description of the process flow and the utilized device equipment.
An efficiently automated production plant represents a basis for the transition from the traditional batch- to the continuous pharmaceutical production. Accurate process monitoring and real-time tracking of essential process quantities are two crucial requirements for the plant efficiency. These requirements are addressed via application of the adequate PAT equipment. Section 2.2 provides a brief theoretical overview, including the FGA definition and PAT framework application procedure. Furthermore, an overview of within the continuous tablet manufacturing plant utilized PAT equipment is provided in this Section.
Fig. 2.1 depicts a schematic structure of the continuous tablet manufacturing plant, including the individual unit operations and the employed PAT equipment.

2.1 Description of Process Flow

2.1.1 HME Line

The HME line involves several individual unit operations:

- ***Feeding***: Two loss-in-weight feeders, namely, Brabender MTS-Hyg (Brabender, Germany) and K-TRON KT20 (Coperion K-Tron, Switzerland), from now on referred to as API and polymer feeder. Feeders are equipped with mass scales and operate in the gravimetric regime, i.e. adjust the screw speed based on the loss-in-weight measurements in order to reach the desired mass flow.
- ***Extrusion***: Twin screw extruder Coperion ZSK 18 (Coperion, Germany).
- ***Cooling Track***: Conveyor belt and proportional pressure regulator VPPE-3–1–1 (FESTO, Germany).
- ***Pelletisation***: Pelletisation system PRIMO 60E (Maag, Germany).
- ***Discharge***: Pneumatic discharge valve.
- ***Pneumatic transport***: Vacuum transport device Piovan S50 (Piovan, Italy).

API and polymer feeder are placed on the top of the HME line. They are manually filled with the API and polymer powder, which they supply to to the extruder inlet. Inside the extrusion unit, polymer powder melts and API is distributed in the polymer matrix. Fed powder material changes its structure, and the outcome of the extrusion unit is one strand. Produced strand is transported among the cooling track via conveyor belt. A pressure regulator is placed above the cooling track and cools the transported strand by means of the pressurized air. Finally, the cooled strand enters the pelletisation unit, where it is cut in the cylindrical pellets. Produced pellets are gathered in a canonical hopper, which is connected to a pneumatic discharge valve and a vacuum transport device. If the pellets meet the predefined expectations regarding their API content, they can be transported to the DC line via vacuum transport device. Otherwise, the hopper is emptied via discharge valve. Table 2.1 provides an overview of the available process data and nominal operating points of the individual unit operations in the HME line.

Table 2.1: HME line: Process data and nominal operating points of the individual unit operations.

Process Unit	**Process Data**	**Nominal operation point**
Feeding	Hold-up Screw speed Mass flow	Mass flow $\approx 2kg/h$
Extrusion	Barrel temperature Screw speed Melt pressure	$70 - 130°$ $200rpm$
Cooling	Air pressure	$p_{air} \approx 3bar$
Pelletisation	Intake speed	$v \approx 1.293\frac{m}{min}$
Discharge	Activation signal	$d_{HME} = 1$ (active)
Transport	Activation signal	$t_{HME} = 0$ (inactive)

2.1.2 DC Line

DC line incorporates several individual unit operations, as well:

- ***Feeding***: Two loss-in-weight K-TRON KT20 (Coperion K-Tron, Switzerland) feeders, from now on referred to as pellet and pre-mix feeder. Again, feeders are equipped with mass scales and operate in the gravimetric regime.
- ***Blending***: Hosokawa Modulomix (Hosokawa Micron, Netherlands).
- ***Discharge***: Pneumatic discharge valve.
- ***Tabletting***: Fette 102i (Fette, Germany).

A starting point of the DC line is a feeding-blending-unit (FBU), which consists of the pellet- and pre-mix feeder, and a blender. This unit is placed on the platform

Table 2.2: DC Line: Process data and nominal operation points of the individual unit operations.

Process Unit	Process Data	Nominal point
Feeding	Weight Screw speed Mass flow	Mass flow $\approx 5kg/h$
Blending	Screw speed	$n_B \approx 800rpm$
Discharge	Activation signal	$d_{DC} = 1$ (active)
Tabletting	Turret speed Compaction parameters	$n_{TP} \approx 41.67rpm$

above the TP, on the top of the DC line. Pellet feeder is supplied with the pellets produced within the HME line. Pre-mix feeder is manually filled with the pre-mix powder material. Feeders supply pellets and powder to the blender, which blends these two materials in the liquidizing regime. Quality of the resulting material is examined at the blender outlet. If the pellets/powder mixture satisfies the predefined expectation regarding the API content, material can be proceeded to the TP and compressed to the tablets. Otherwise, material is redirected to the waste hopper via discharge flap. Table 2.2 provides an overview of the available process data and nominal operating points of the individual unit operations in the DC line.

2.2 Process-Analytical-Technology (PAT)

Process analytical technology (PAT)[1]*:*
"The FDA agency considers PAT to be a system for designing, analyzing, and controlling manufacturing through timely measurements (, i.e. during processing) of critical quality and performance attributes of raw and in-process materials and processes, with the goal of ensuring final product quality."

PAT equipment contributes to the improved process understanding and control of the continuous manufacturing plants. PAT is utilized for the process monitoring via data acquisition, as well as for the development of the risk-mitigation strategies. Additionally, PAT supports the effective control of the critical quality attributes (CQA).
Design and optimization of drug formulations and manufacturing processes within the PAT framework typically include the following steps:

1. Identification and measurement of the critical material and process quantities related to the product quality

[1] *Guidance for Industry PAT*, A Framework for Innovative Pharmaceutical Development, Manufacturing and Quality Assurance

2. Design of the (near) real-time measurement system for the monitoring of the critical quantities addressed in the previous point
3. Development of the appropriate concepts to control the critical attributes
4. Determination of mathematical relations between the product quality attributes, and critical material and process quantities

Note: Sequence of these steps can vary from process to process.
Via provided real-time process monitoring and continuous tracking of the product quality, process quantities of interest are gathered and forwarded to the control concepts for further action. In such a way, PAT equipment may be perceived as an important support of the control strategies in the continuous manufacturing.
It is of great importance to recognize and consider the process quantities, which have a significant influence on the quality of the intermediate and final product. First of all, API concentration among the continuous plant should be tracked, as it has the crucial impact on the uniformity of intermediate and final product. Appropriate PAT equipment for the measurement of the API concentration is employed and concentration is measured by means of the near infrared spectroscopy (NIR:Sentronic SentroPAT FO NIR Type 1100/2200/256/4). NIR device can be placed in the HME or in the DC line, i.e. at the extruder or blender outlet, respectively. The utilized NIR device allows the quality tracking among the continuous plant. The acquired data can be utilized for modeling and control purposes.
Quality of within the HME line produced pellets is affected by two process quantities, i.e. strand diameter and temperature. 3 axis laser and infrared sensor are employed for the measurement of these quantities.
Another significant process quantity is a fill level in the TP hopper. Previous researches have shown, that the large deviations from the nominal points, can lead to the quality degradation of the produced tablets. Again, the appropriate PAT equipment for tracking this process quantity needs to be utilized. Fill level measurement takes place via Ultra Sonic Sensor.
Table 2.3 provides an overview of the utilized PAT equipment within the investigated continuous tablet manufacturing plant.
Process- and PAT equipment are connected to a XAMControl automation software [evon, 2020] via following data exchange protocols:

1. API, pre-mix and pellet feeder are connected via PBpro ETH remote Profibus interface (Softing GmbH, 2019).
2. Polymer feeder is connected via Modbus TCP interface.
3. Vacuum transport is connected via Modbus RTU interface.
4. Extruder is connected via OPC UA.
5. TP and NIR spectrometer are connected via UPC DA.
6. 3-axis measurement device is connected via Ethernet TC/TP.
7. Pelletizer, blender, discharge units, IR-sensor, US-sensor and pressure regulator are connected via B&R Bus Controller and analog I/O modules (B&R Industrial Automation, 2019).

Tasks related to the data acquisition and analysis, and the required user-device communication, are implemented by means of the XAMControl Automation Software platform. XAMControl offers a process visualization and provides a database for the real time storage of data. For modeling purposes required laboratory experiments are programmed and executed in the XAMControl. The experimental

data is collected and stored for the post processing. Control concepts are also implemented within this software.

PAT Equipment			
Nr.	**Device**	**Position**	**Measurement**
1	NIR Spectrometer	Extrusion Inlet	Concentration
2	3axis Laser	Pelletisation Inlet	Diameter
3	Infrared Sensor	Pelletisation Inlet	Temperature
4	NIR Spectrometer	Blender Outlet	Concentration
5	Ultra Sonic Sensor	TP hopper	Fill level

Table 2.3: Overview of within the continuous tablet manufacturing plant utilized PAT Equipment, including the utilized devices, measurement position and measured process quantities. The introduced numbering corresponds to the numbering depicted in the Fig. 2.1

2.3 Schematic Overview of Continuous Tablet Manufacturing Plant

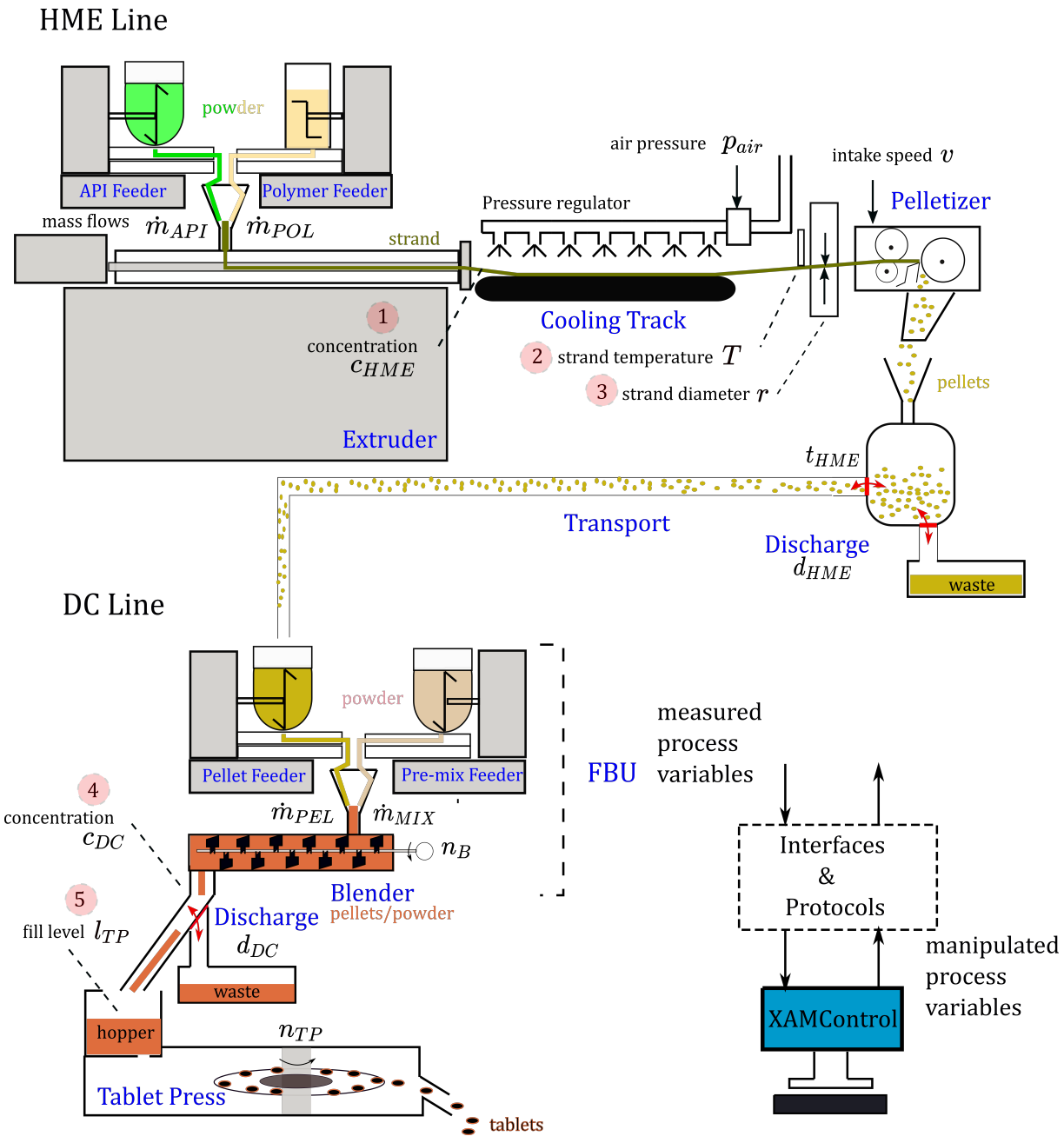


Figure 2.1: Schematic overview of the continuous tablet manufacturing plant: HME and DC line involve several individual unit operations depicted with the blue letters. Different material structures are color distinguished, i.e powder → strand → pellets → pellets/powder → tablets. Via PAT equipment monitored measurements are indicated with the light red circles.

3 Process Modeling

Model Identification[1]*:*
"A modeling procedure can be summarized in the following manner: The real system or process is replaced with a simple, yet adequate model. A model reflects the functional relations between the essential system quantities, and captures the behavior of a real system."

Aim of a modeling task is a development of simple, yet adequate description of a system or process of interest. Starting point of this task is to determine and take into account the process quantities/variables/signals, which have a significant impact on the process behavior. Outcome of modeling are the mathematical relations between the essential system quantities.
A typical modeling approach involves a decomposition of one complicated system in the simpler, individual subsystems. In such a way, essential system quantities can be recognized in the simple manner, and the required mathematical relations can be determined. In that sense, the investigated system of interest, i.e. the continuous tablet manufacturing plant, is decomposed in the HME and DC line, and the lines are decomposed in the individual modeling units. DC line involves three modeling units, namely, blender concentration, blender mass flow and tablet press modeling unit. HME line involves extrusion concentration, extrusion mass flow and pelletisation modeling unit. Sections 3.2 and 3.3 provide an overview of the model identification for the DC and HME line, respectively.
Process quantities of interest can be divided in the two categories, input and output signals. Input signals can be perceived as the process actuators. These are the quantities which affect the system and can be externally manipulated. On the other hand, output signals are the process quantities, which can be captured. These signals can be perceived as the system reaction to the applied input signals. Input and output signals of the individual modeling units in the DC and HME line are presented in the Tables 2.1 and 2.2, respectively.
Within the course of the thesis, a process modeling is carried out via data-driven model identification, by means of the MATLAB System Identification Toolbox App and MATLAB function *ssest*. Subsections 3.1.1 and 3.1.2 provide a brief overview of these two identification approaches. Additionally, the Subsections 3.2.1 and 3.2.2 provide a step-by-step toolbox application on the example of blender concentration and blender mass flow modeling unit.

[1] [Horn and Dourdoumas, 2004], Regelungstechnik: Systeme und deren Beschreibung

3.1 Data-driven Modeling via System Identification Toolbox

This Section introduces two data-driven modeling approaches. Data-driven modeling implies a strong dependency of the model quality on the provided experimental data. In that sense, the identification experiments need to be designed carefully. A complete operating range of interest should be covered, i.e. the designed input- and collected output signals should precisely reflect the system behavior in this range. Although the model identification process itself does not include the information about the physical behavior of the system, this knowledge can be utilized by the design of identification experiments. Model order is user defined and chosen as a compromise between the model complexity and accuracy. Fig. 3.1 illustrates the measurement configuration for collecting the input- and output sequences required for the identification procedure. The input- and output sequences are uniformly sampled with sampling time T_s and imported to the System Identification Toolbox.

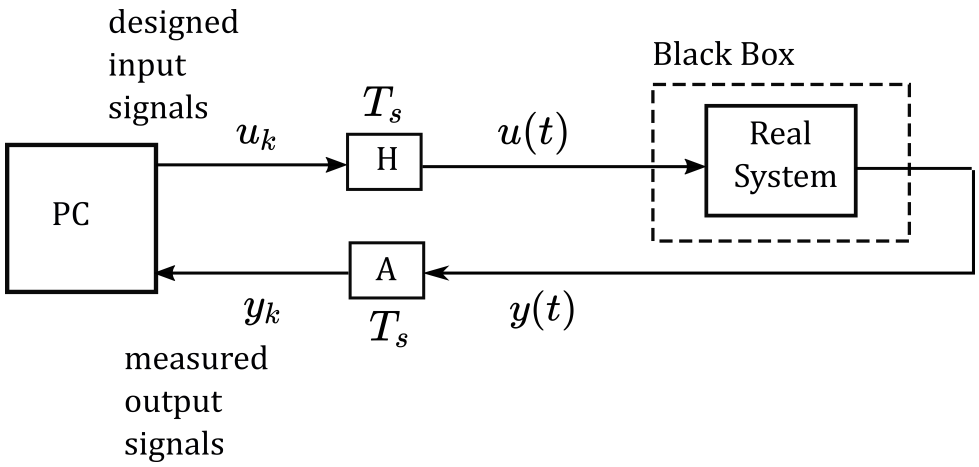


Figure 3.1: Data-driven modeling measurement configuration: Input signals are designed and sent to the real system via process actuators. Output signals are measured via process sensors. Input- and output sequences are collected and provided to the System Identification Toolbox.

3.1.1 System Identification Toolbox App

The System Identification Toolbox App in MATLAB [SysIdTool, 2020] provides an application for the data-driven modeling of dynamic systems. Among others, it offers a feature for the identification of time-continuous or time-discrete transfer functions using the time- or frequency domain data as input- and output signals. The performed identification procedure can be summarized in the following manner:

- Data preparation, i.e. carrying out the required experiments and pre-processing of the collected experimental data
- Data import, i.e. providing the toolbox with input- and output signals collected and prepared in the previous point

- Structure definition, i.e. specifying the transfer function structure (number of poles, number of zeros, etc.)
- Parameter identification, i.e. estimating the adjustable parameters of transfer function (numerator and denominator coefficients, delay time)
- Model validation, i.e. a comparison between the measured- and model-estimated output signals

The outcome of the identification procedure is a plant transfer function $P(s) = \frac{Y(s)}{U(s)}$, with $U(s)$ and $Y(s)$ denoting the Laplace transforms of input and output signal, respectively. Transfer function representation utilizes Laplace transformation and maps the relations from time- to the frequency domain. In that sense, $P(s)$ relates the input- and output signal of the investigated system in the complex variable s domain. This representation is particularly suitable for the SISO systems, as well as for the development of standard control concepts, such as algebraic synthesis or PID control. Therefore, this identification approach is utilized for the concentration modeling.

3.1.2 System Identification Toolbox: *ssest* Function

This Section outlines another approach for the data-driven estimation of model parameters by means of the MATLAB function *ssest* [ssest, 2020]. As in the previously introduced identification approach, time- or frequency domain experimental data should be collected, pre-processed and imported to the toolbox.
One possible outcome of the identification procedure is a system of ordinary differential equations first order (ODE 1st), from now on referred to as a time-continuous state-space-model. This model reads as:

$$\frac{dx(t)}{dt} = A \cdot x(t) + B \cdot u(t) + B_d \cdot d(t) \quad \text{and} \quad y(t) = C \cdot x(t) + D \cdot u(t) + D_d \cdot d(t)$$

Input-, output- and disturbance signals, symbolized with $u(t)$, $y(t)$ and $d(t)$, respectively, are provided from the identification experiments. $x(t)$ depicts a state vector with n states. Constant size matrices A, B, B_d, C, D and D_d are the model parameters which need to be estimated within the identification procedure.
Alternatively, outcome of the identification procedure can be a system of linear constant equations relating the state variables in the consecutive time points, from now on referred to as a time-discrete state-space-model. This model reads as:

$$x_{k+1} = A_D \cdot x_k + B_D \cdot u_k + B_{Dd} \cdot d_k \qquad \text{and} \qquad y_k = C_D \cdot x_k + D_D \cdot u_k + D_{Dd} \cdot d_k$$

Notation of the introduced signals, as well as of the system parameters, stays the same as in the time-continuous representation. Matrices, i.e. to be estimated model parameters, are symbolized with sub-index D. Signals are evaluated at the time instance kT_s, and therefore denoted with the sub-index k. State-space-model representation is particularly suitable for modeling the MIMO systems, such as blender mass flow, TP, etc.
Identification procedure requires several user defined arguments, such as the sampling time, feedthrough, estimation options, etc. Model order can be specified by user, or automatically estimated by a solver. Following lines provide a code example for the implementation of this identification approach in MATLAB. The

outcome of introduced procedure is a time-discrete state-space model, which can be utilized for the MPC design in the straightforward manner. On the other hand, the time-continuous representation would require the additional discretization effort, and therefore is omitted.

```
% Example of ssest Function Identification Procedure: System with 2 input- and 2 output signals is investigated

% Step 1: Import the experimental data and specify sampling time
model = iddata([y1, y2],[u1, u2], Ts);
model.InputName = {'Input 1-Name','Input 2-Name'};
model.InputUnit = {'Input 1-Unit','Input 2-Unit'};
model.OutputName = {'Output 1-Name','Output 2-Name'};
model.OutputUnit = {'Output 1-Unit','Output 2-Unit'};
model.ExperimentName = {'Experiment-Name'};

% Step 2: Specify model properties
model_order = n_model;
feedthrough = true;

% Step 3: Specify estimation options:
opt = ssestOptions;
% Enforce stability of estimated model
opt.EnforceStability = true/false;
% Specify initial state
opt.InitialState = '';
% Numerical search method for iterative parameter estimation
opt.SearchMethod = '';
% Specify error to be minimized
opt.Focus = 'simulation';
% Specify forward- and backward prediction horizons
opt.N4Horizon = [r sy su];

% Step 4: Estimate model parameters
model_est = ssest(model, model_order, 'Ts', Ts, 'Feedthrough', feedthrough, 'DisturbanceModel', 'none', opt);
figure; step(model_est); % step response

% Step 5: Validation of estimated model
figure; compare(model, model_est);
```

Note: The introduced MATLAB code can be extended, and further estimation options can be specified. However, this code represents a sufficient basis for the identification procedure.

3.2 Modeling of DC Line

DC Line consists of three individual modeling units, namely, blender concentration-, blender mass flow- and TP modeling unit.
Subsection 3.2.1 outlines the identification of the transfer function relating the concentrations at the blender inlet and the blender outlet. Additionally, this Subsection provides a step-by-step System Identification Toolbox App application procedure on this example. The identified transfer function is utilized for the design and tuning of the concentration control concept.
Subsection 3.2.2 holds the identification of the state-space-model relating the blender inlet mass flow and blender rotational speed as the input-, and the blender outlet mass flow and hold-up as the output variables for this modeling unit. A state-space-model relating the TP inlet mass flow and turret speed as input-, and tablet press fill level as output variable is identified within the Subsection 3.2.3. State-space-modeling is performed by means of in the Subsection 3.1.2 introduced identification approach.
Performed modeling is strongly control-oriented, i.e. the input- and output variables of the individual modeling units are potential candidates for the manipulated and controlled variables. Yet, not all input variables of the individual modeling units, e.g. TP inlet mass flow, can be directly manipulated in the continuous plant operation. On the other hand, some output variables, e.g. blender outlet mass flow and hold-up, cannot be measured in the continuous operation, as well. Therefore, the identified blender and TP state-space-model are joined to an united state-space-representation with blender inlet mass flow and TP turret speed as input-, and TP fill level as the output variables. The resulting representation, from now on referred to as DC line model, is developed within the Subsection 3.2.4. This combined representation of DC line is particularly suitable for the development of TP control concept in the further course of the thesis, as the input variables can be manipulated, and the output variable can be measured in the continuous plant operation. Table 3.1 provides a compact overview of the individual modeling units and the associated input- and output variables.

Table 3.1: Overview of the individual modeling units in the DC line

Modeling Unit Unit	**Input Signals**	**Output Signals**
Blender: Concentration	inlet concentration c_{DCin}	outlet concentration c_{DCout}
Blender: Mass flow	inlet mass flow $\dot{m}_{Bin}$ turret speed n_B	outlet mass flow $\dot{m}_{Bout}$ hold-up m_{Bhu}
TP	inlet mass flow $\dot{m}_{TPin}$ turret speed n_{TP}	hopper fill level l_{TP}
DC Line	inlet mass flow $\dot{m}_{Bin}$ turret speed n_{TP}	hopper fill level l_{TP}

3.2.1 Modeling of Blender Concentration

Blender concentration transfer function relates the concentrations at the blender inlet and blender outlet. According to [Kruisz et al., 2017], blender inlet concentration corresponds to:

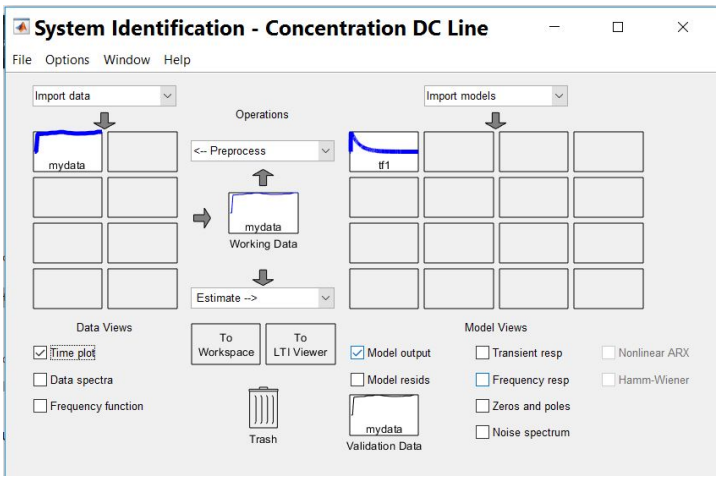
$$c_{DCin} = \frac{\dot{m}_{PEL}}{\dot{m}_{PEL} + \dot{m}_{MIX}}$$

In that sense, the desired inlet concentration can be adjusted by manipulating the ratio of pellet- and pre-mix feeder mass flows. Blender outlet concentration c_{DCout} is captured by means of the NIR measurement device placed between the blender outlet and the discharge flap. These quantities can be manipulated/measured in the continuous plant operation.
The experimental data required for the identification purposes is collected via laboratory experiment depicted in the Fig.3.3. Blender inlet concentration is varied from 10% to 35% in 2.5% and 5% steps. The step duration is adjusted, such that the blender outlet concentration reaches the steady state after each change. In such a way, the required dynamic behavior of the investigated system can be captured. The investigated operating range corresponds to the measurement range of NIR device, where the linear system behavior is expected.
The input- and output sequences are collected and imported to the System Identification Toolbox App. The structure of transfer function is specified based on the structure proposed in [Kruisz et al., 2017], i.e. a transfer function containing one zero, three poles and a delay-time term. Step-by-step toolbox application procedure is illustrated in the Fig. 3.2. Outcome of the identification procedure is a proper transfer function third order, from now on referred to as DC concentration transfer function:

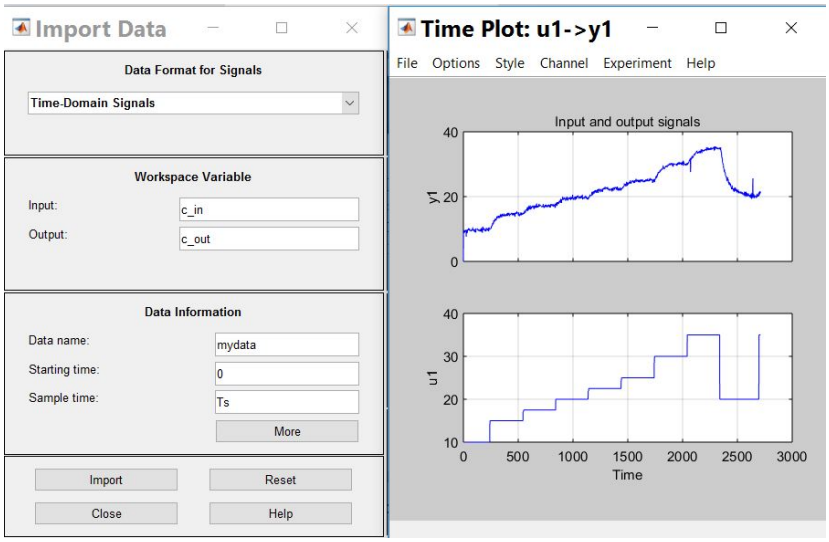
$$P_{C,DC}(s) = \frac{C_{DCout}(s)}{C_{DCin}(s)} = e^{-sT_{d,DC}} P^{\star}_{DC}(s)$$

$$\text{with} \qquad P^{\star}_{DC}(s) = \frac{0.03s + 2.48 \cdot 10^{-5}}{s^3 + 1.86s^2 + 0.03s + 2.47 \cdot 10^{-5}} \qquad \text{and} \qquad T_{d,DC} = 7s$$

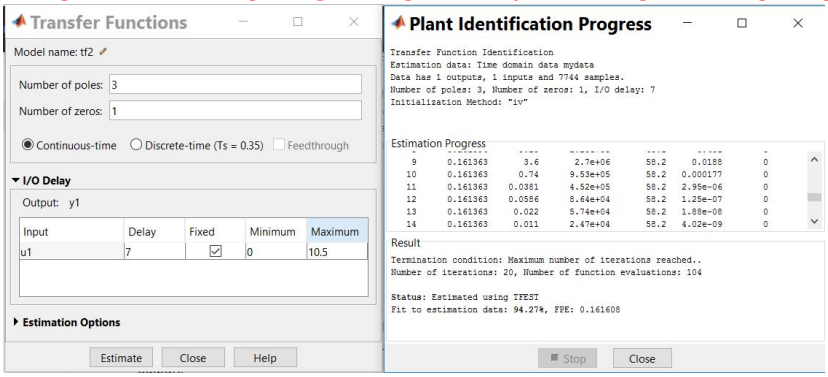
Laboratory experiments have shown, that the outlet concentration equals the inlet concentration after a certain time interval. There are two factors, which have an influence on the duration of this time interval, i.e. a blender speed n_B and blender residence time distribution (RTD), i.e. the time one particle spends passing among the blender.
Fig. 3.3 illustrates the set inlet concentration c_{DCin}, measured outlet concentration c_{DCout} and via identified transfer function estimated outlet concentration $c_{DCout,sim}$. The obtained results indicate a very good conformity between the measured and model estimated signals, i.e. between the real system and the identified transfer function. The identified transfer function is utilized for the design of DC line concentration controller, as well as for the discharge signal triggering, in the simulation.



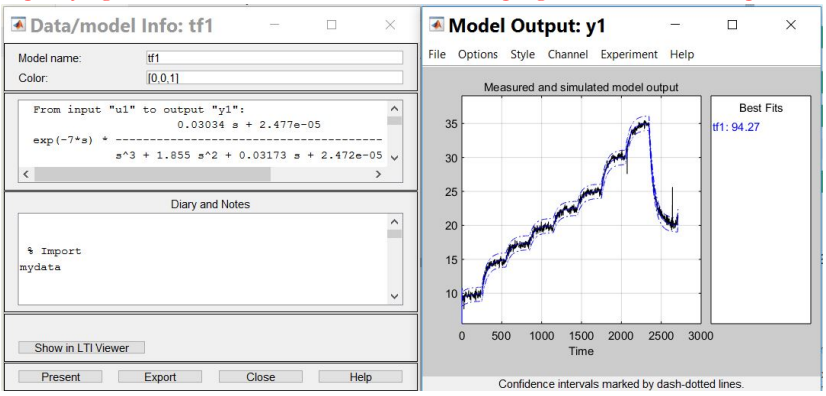
(a) Step 1: System Identification Toolbox App user interface



(b) Step 2: Import (left) and time-plot (right) of experimentally collected input- and output sequences



(c) Step 3: Specifying the transfer function structure (left) and progress of identification procedure (right)



(d) Step 4: Resulting transfer function (left) and model validation (right)

Figure 3.2: Transfer function identification via System Identification Toolbox App in MATLAB

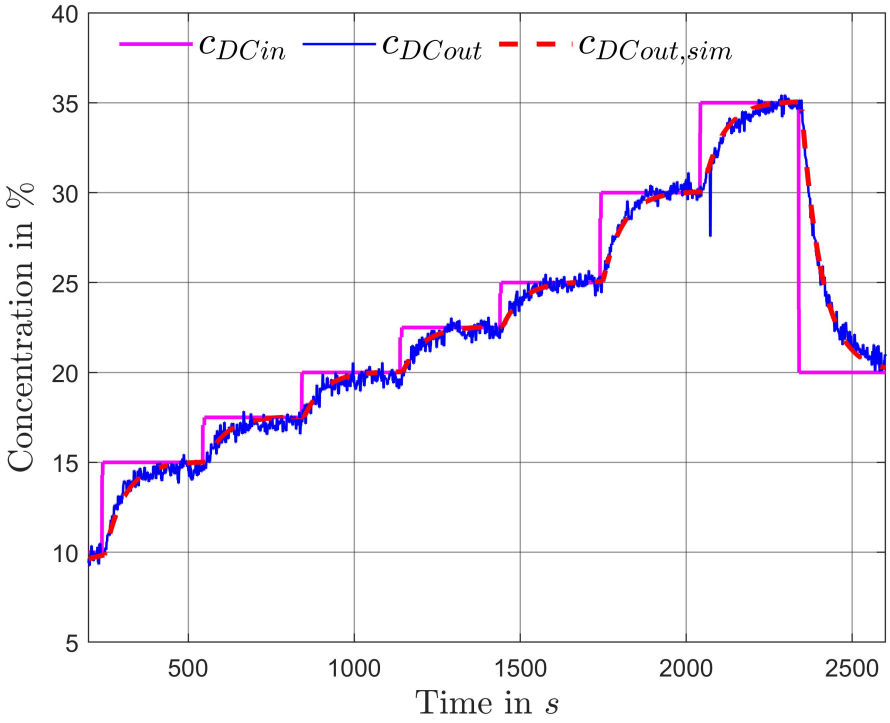


Figure 3.3: Laboratory Experiment-Identification of blender concentration model: Magenta line indicates the set inlet concentration, blue line indicates the NIR measured outlet concentration, and dashed red line indicates via identified transfer function estimated outlet concentration.

3.2.2 Modeling of Blender Mass Flow

This Subsection outlines the identification of a blender state-space-model relating the blender outlet mass flow and hold up as output-, and blender inlet mass flow and rotational speed as input signals.
[Rehrl et al., 2016] suggests that, besides the data-driven approach, the investigated blender unit can be modeled by means of the physical laws describing the blender hold-up. However, the outcome of this approach is a non-linear model. This would lead to the additional linearization efforts, as the standard MPC procedure requires the linear system representation. On the other hand, data-driven approach assumes a linear system behavior in a certain range around the nominal operating points. Outcome of this approach is a linear model, which can be utilized by the controller in a straightforward manner. Due to that, the physical system modeling is omitted. As the time-discrete state-space representation is particularly suitable for the intended MPC design, the in the Subsection 3.1.2 introduced modeling approach is chosen. Yet, for the application of standard control techniques, blending unit can be modeled via transfer functions, as suggested in the Subsection 3.1.1. Further details on that kind of modeling procedure can be found in [Celikovic et al., 2019]. Blender inlet mass flow corresponds to the sum of pellet and pre-mix feeder mass flow, i.e. $\dot{m}_{Bin} = \dot{m}_{PEL} + \dot{m}_{MIX}$. Blender rotational speed is set by an internally

integrated speed controller. Both of these quantities can be manipulated in the continuous plant operation.
Blender outlet mass flow is indirectly measured via custom-built catch scale (HBM PW22C3 single point load cell), placed at the blender outlet. Outlet mass flow $\dot{m}_{Bout}$ is calculated using the scale signal m_{Bout}, as:

$$\dot{m}_{Bout}(t) = \frac{dm_{Bout}(t)}{dt} \approx \frac{m_{Bout}(t) - m_{Bout}(t - \Delta t)}{\Delta t} \qquad \text{with} \quad \Delta t = T_s$$

As a consequence of high sampling rate ($T_s = 350ms$) and noise-amplifying characteristics of derivative operator, the resulting signal cannot be employed for the model identification purposes. In order to attenuate the high-frequency components, evaluated signal is pre-processed with low-pass zero-phase forward-backward filter. This class of filter does not shift the signal and invokes no time delay. Best filtering results are obtained for the cut-off frequency $f_{cut} = 0.05Hz$ and filter order 32.
Blender hold-up is calculated via simple mass-balance equations, as:

$$\begin{aligned} m_{Bin,k} &= m_{PEL,k} + m_{MIX,k} && \text{total mass of material in feeders} \\ \Delta m_{Bin,k} &= m_{Bin,0} - m_{Bin,k} && \text{amount of fed material} \\ \Delta m_{Bout,k} &= m_{Bout,k} - m_{Bout,0} && \text{amount of blended material} \\ m_{Bhu,k} &= \Delta m_{Bin,k} - \Delta m_{Bout,k} && \text{blender hold-up} \end{aligned}$$

Note: The catch scale is brought into the device setup solely for the modeling purposes. It cannot be placed at the blender outlet during the continuous plant operation. In that sense, the blender outlet mass flow and consequently the hold-up, cannot be measured in the continuous plant operation, as well.
Experimental data required for the identification purposes is collected via in the Fig. 3.4 depicted laboratory experiment. In the first part of the experiment ($t \leq 1000s$), blender rotational speed is kept constant and the blender inlet mass flow is varied around the nominal point in the $\pm 2.5kg/h$ steps. In the second part of the experiment ($t > 1000s$), the inlet mass flow is kept constant, and the blender rotational speed is varied around the nominal point in the $\pm 200rpm$ steps. In such a way, impact of the independent system inputs, can be captured.
Blender rotational speed and inlet mass flow, i.e. its deviation from the nominal operating points, are imported as the input sequences, and blender outlet mass flow and hold-up, i.e. its deviation from the nominal operating points, are imported as the output sequences to the System Identification Toolbox. The best identification results are obtained by utilizing the model specifications and estimation options introduced in the following MATLAB code.

```
% Blender Modeling
% Step 1
blender = iddata([dot_m_Bout, m_Bhu],[n_B, dot_m_Bin], Ts);
blender.InputName = {'rotational speed','inlet mass flow'};
blender.InputUnit = {'rpm','kg/h'};
blender.OutputName = {'outlet massflow','hold-up'};
blender.OutputUnit = {'kg/h','g'};
blender.ExperimentName = {'Identification_experiment'};

% Step 2
model_order = 3;
feedthrough = true;

% Step 3
opt = ssestOptions;
opt.EnforceStability = true;
opt.InitialState = 'zero';
% gna - adaptive subspace Gauss-Newton search
opt.SearchMethod = 'gna';
% error between measured and simulated outputs is minimized
opt.Focus = 'simulation';
% chosen via trial and error attempts
opt.N4Horizon = [10 3 2];

% Step 4
blender_est = ssest(blender, model_order, 'Ts', Ts, '
    Feedthrough', feedthrough, 'DisturbanceModel', 'none',
    opt);
figure; step(blender_est);

% Step 5
figure; compare(blender, blender_est);
```

Iterative parameter identification is performed via adaptive subspace Gauss-Newton numerical search method. Model parameters, A_B, b_{B1}, b_{B2}, C_B, d_{B1} and d_{B2}, are identified, such that the simulation error between the measured and simulated outputs is minimized. Model order is chosen as a compromise between the model accuracy and complexity. Outcome of the identification procedure is a time-discrete state-space-model third order, from now on referred to as blender model, which reads as:

$$\begin{aligned} x_{B,k+1} &= A_B x_{B,k} + b_{B1}\Delta u_{B1,k} + b_{B2}\Delta u_{B2,k} \\ \Delta y_{B,k} &= C_B x_{B,k} + d_{B1}\Delta u_{B1,k} + d_{B2}\Delta u_{B2,k} \end{aligned} \tag{3.1}$$

with: $u_B = [u_{B1} \quad u_{B2}]^T = [n_B \quad \dot{m}_{Bin}]^T$ and $\Delta u_B = u_B - u_{B,nom}$

$y_B = [y_{B1} \quad y_{B2}]^T = [\dot{m}_{Bout} \quad m_{Bhu}]$ and $\Delta y_B = y_B - y_{B,nom}$

Nominal operating points of blender speed, blender inlet mass flow, blender outlet mass flow and blender hold-up are $n_{B,nom} = 800rpm$, $\dot{m}_{Bin,nom} = 5kg/h$, $\dot{m}_{Bout,nom} = 5kg/h$ and $m_{Bhu,nom} = 58g$, respectively.

Fig. 3.4 depicts all signals of interest during the executed laboratory test. The obtained results indicate a very good conformity between the measured and model estimated output signals, and verify the quality of the identified blender model.

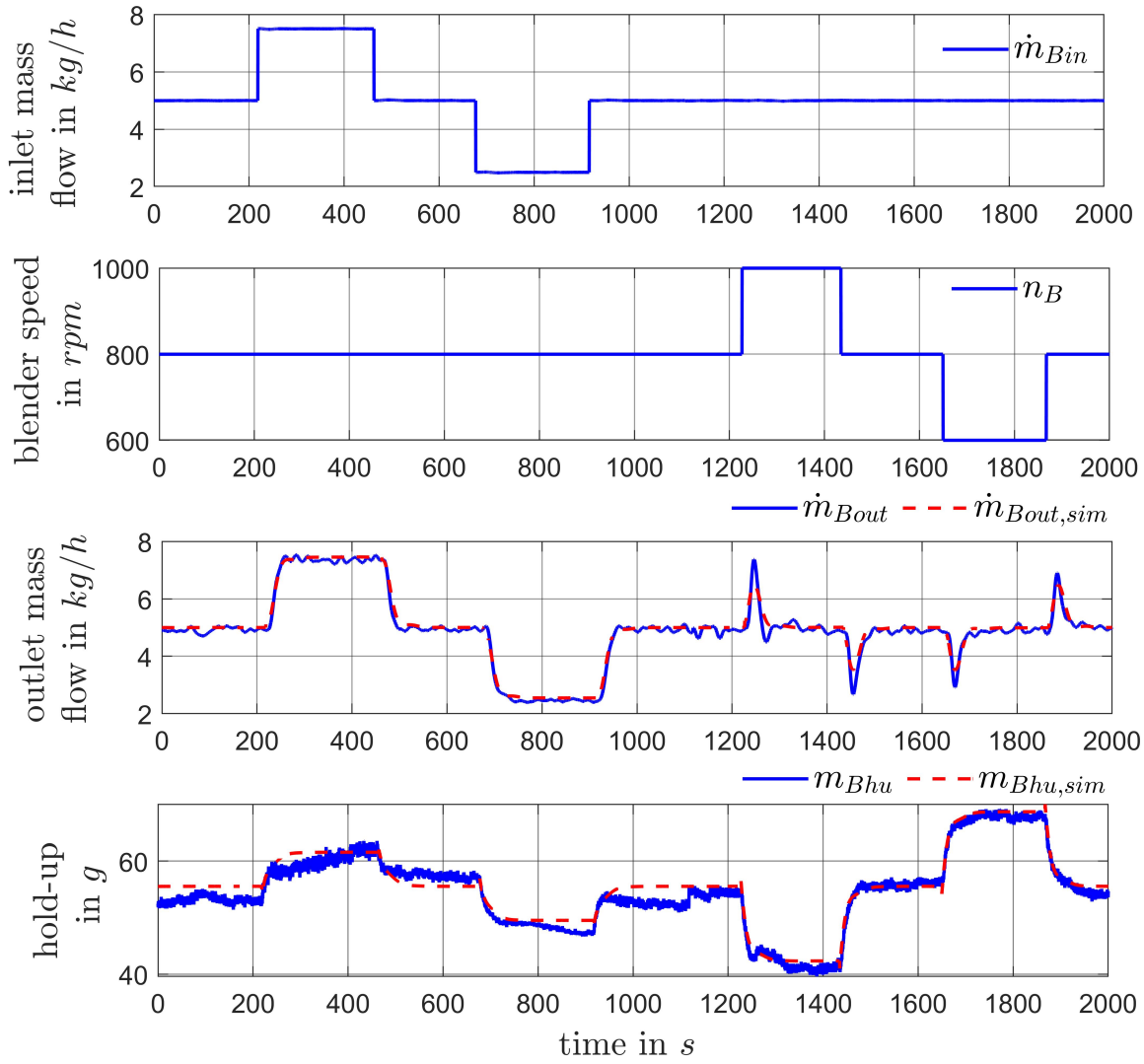


Figure 3.4: Laboratory Experiment-Identification of blender mass flow model: First and second subplot depict the set blender inlet mass flow and rotational speed, respectively. Third subplot depicts the measured- and via identified model estimated outlet mass flow in blue and red color, respectively. Fourth plot depicts the measured- and via identified model estimated blender hold-up in blue and red color, respectively.

3.2.3 Modeling of Tablet Press

This Subsection outlines a development of a simple state-space-model first order, relating the TP fill level as output-, and TP inlet mass flow and turret speed as the input signals.
Model follows the mass balance equation describing the TP hopper hold-up $\dot{m}_{TPhu} = \dot{m}_{TPin} - \dot{m}_{TPout}$, and the relation between the hopper outlet mass flow and TP turret speed $\dot{m}_{TPout} = \gamma n_{TP}$. Furthermore, TP fill level is proportional to the TP hopper hold-up $l_{TP} \sim m_{TPhu}$.
Experimental data required for the identification purposes is collected via laboratory experiment depicted in the Fig. 3.5. The material is fed directly to the TP hopper, and the desired TP inlet mass flow is set by adjusting the feeder mass flow. However, in the continuous plant operation, the inlet mass flow cannot be directly manipulated, as it corresponds to the blender outlet mass flow. TP fill level is measured via Ultra-Sonic-Sensor placed in the TP hopper. In the first part of the experiment, inlet mass flow is kept constant and turret speed is varied around the nominal operating point in $\pm 20 rpm$ steps. Thereafter, turret speed is kept constant and the inlet mass flow is varied around the nominal operating point. In such a way, the influence of the independent inputs, can be captured.
TP turret speed and inlet mass flow, i.e. its deviation from the nominal operating points, are imported as the input sequences, and TP fill level, i.e. its deviation from the nominal operating point, is imported as the output sequence to the System Identification Toolbox. Model specifications are set based on the known physical properties of the system, model order is set to one and feedthrough to false. The remaining estimation options remain the same, as in the MATLAB code introduced in the Section 3.2.2. Again, iterative parameter identification is executed and the error between the measured and simulated outputs minimized. Outcome of the identification procedure is a time-discrete state-space-model first order, from now on referred to as TP model, which reads as:

$$\begin{aligned} x_{TP,k+1} &= A_{TP} x_{TP,k} + b_{TP1} \Delta u_{TP1,k} + b_{TP2} \Delta u_{TP2,k} \\ \Delta y_{TP,k} &= c_{TP}^T x_{TP,k} \end{aligned} \tag{3.2}$$

$$\text{with:} \quad u_{TP} = [u_{TP1} \quad u_{TP2}]^T = [n_{TP} \quad \dot{m}_{TPin}]^T \quad \text{and} \quad \Delta u_{TP} = u_{TP} - u_{TP,nom}$$
$$y_{TP} = l_{TP} \quad \text{and} \quad \Delta y_{TP} = y_{TP} - y_{TP,nom}$$

Nominal operating points of TP turret speed, TP inlet mass flow and TP fill level are $n_{TP,nom} = 41.67 rpm$, $\dot{m}_{TPin,nom} = 5 kg/h$ and $l_{TP} = 0.15 m$, respectively. Fig. 3.5 depicts all signals of interest during the executed laboratory experiment. The obtained results indicate a very good conformity between the measured and model estimated TP fill level. In that sense, a quality of the identified TP model is confirmed, as well.

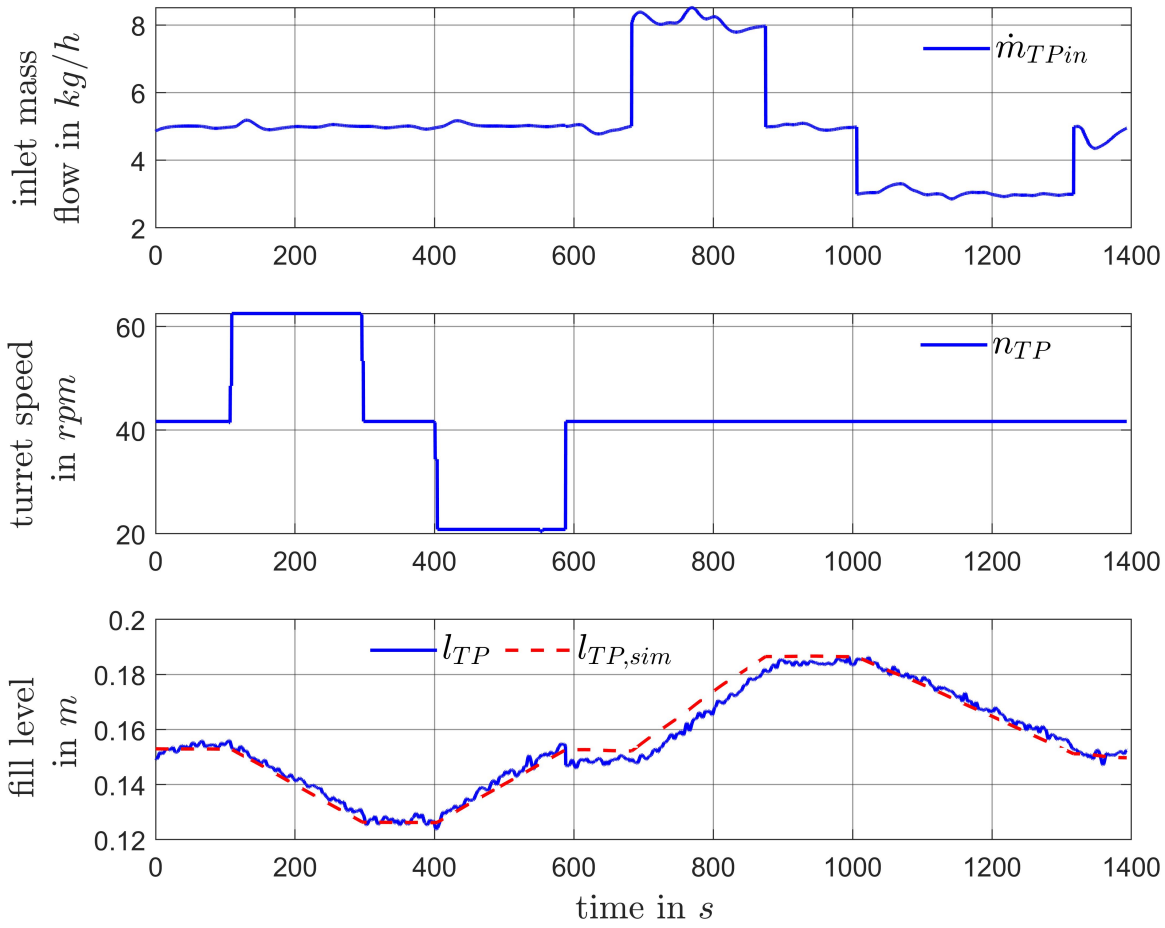


Figure 3.5: Laboratory Experiment-Identification of TP model: First and second subplot depict the set TP inlet mass flow and turret speed, respectively. Third subplot depicts the measured- and via identified model estimated TP fill level outlet mass flow in blue and red color, respectively.

3.2.4 DC Line Model

As introduced in the Subsections 3.2.2 and 3.2.3, not all input/output variables of the individual modeling units can be manipulated/measured in the continuous plant operation. This Section outlines a development of a DC line model, which assemblies the previously identified blender and TP model. As it is not possible to directly manipulate the TP inlet mass flow, but rather to affect it with a change of the blender inlet mass flow, the resulting DC model utilizes the blender inlet mass flow as the input signal. The resulting DC line state-space-representation can be employed for the development of TP fill level controller. It is experimentally proven that the blender rotational speed has neglectable influence on the TP fill level, and therefore is set to the nominal operating point. Blender hold-up is not

considered as well. Blender state-space-representation reduces to:

$$x_{B,k+1} = A_B x_{B,k} + b_{B2}\Delta m_{Bin,k} \tag{3.3}$$
$$\Delta m_{Bout,k} = c_{B,1}^T x_{B,k} + d_{B2,1}\Delta m_{Bin,k} \tag{3.4}$$
$$\tag{3.5}$$

with $c_{B,1}^T$ denoting the first line of C_B, and $d_{B2,1}$ denoting the first element of d_{B2}. It should be noted, that in the Eq. 3.1 introduced blender model can be utilized for a development of an individual FBU control concept, as suggested in [Celikovic et al., 2019] and [Rehrl et al., 2016]. However, this control concept would require the blender outlet mass flow as the measured process variable, which is not possible in the existing device setup.
Tablet press state-space-representation remains the same:

$$\begin{aligned} x_{TP,k+1} &= A_{TP}x_{TP,k} + b_{TP1}\Delta n_{TP,k} + b_{TP2}\Delta \dot{m}_{TPin,k} \\ \Delta l_{TP,k} &= c_{TP}^T x_{TP,k} \end{aligned} \tag{3.6}$$

TP representation introduced in the Eq. 3.6 can be modified according to the two possible scenarios in the continuous plant operation. A discharge flap placed between the blender outlet and TP inlet redirects material flow depending on the API content of the blended material. If the discharge flap is activated, material is redirected to the waste bucket and no material enters the TP. Otherwise, the blended material proceeds to the TP hopper and TP inlet mass flow corresponds to the blender outlet mass flow. In that sense, TP model equations can be modified as:

$$\begin{aligned} d_{DC} &= 1 \\ \dot{m}_{TPin} &= 0 \longrightarrow \Delta \dot{m}_{TPin} = -\dot{m}_{TPin,nom} \\ x_{TP,k+1} &= A_{TP}x_{TP,k} + b_{TP1}\Delta n_{TP,k} - b_{TP2}\dot{m}_{TPin,nom} \end{aligned} \tag{3.7}$$

$$\begin{aligned} d_{DC} &= 0 \\ \dot{m}_{TPin} &= \dot{m}_{Bout} \\ x_{TP,k+1} &= A_{TP}x_{TP,k} + b_{TP1}\Delta n_{TP,k} + b_{TP2}\Delta \dot{m}_{Bout,k} \\ x_{TP,k+1} &= A_{TP}x_{TP,k} + b_{TP1}\Delta n_{TP,k} + b_{TP2}c_{B,1}^T x_{B,k} + b_{TP2}d_{B2,1}\Delta \dot{m}_{Bin,k} \end{aligned} \tag{3.8}$$

In the Eq. 3.3, Eq. 3.7 and Eq. 3.8 introduced relations can be assembled to:

$$\underbrace{\begin{bmatrix} x_{B,k+1} \\ x_{TP,k+1} \end{bmatrix}}_{x_{DC,k+1}} = \underbrace{\begin{bmatrix} A_B & 0 \\ a_d^T & A_{TP} \end{bmatrix}}_{A_{DC}} \underbrace{\begin{bmatrix} x_{B,k} \\ x_{TP,k} \end{bmatrix}}_{x_{DC,k}} + \underbrace{\begin{bmatrix} b_{B2} & 0 \\ b_d & b_{TP1} \end{bmatrix}}_{B_{DC}} \underbrace{\begin{bmatrix} \Delta \dot{m}_{Bin,k} \\ \Delta n_{TP,k} \end{bmatrix}}_{u_{DC,k}} + \underbrace{\begin{bmatrix} 0 \\ f_d \end{bmatrix}}_{f_{DC}} \tag{3.9}$$
$$\Delta l_{TP,k} = [0^T \quad c_{TP}^T] x_{DC,k} = c_{DC}^T x_{DC,k}$$

Sub-index d denotes the coupling terms dependent on the discharge signal d_{DC}. If discharge is not activated, coupling between the blender and TP model is realized via terms:

$$a_d^T = b_{TP2} \cdot c_{B,1}^T \cdot (1 - d_{DC}) \qquad \text{and} \qquad b_d = b_{TP2} \cdot d_{B2,1} \cdot (1 - d_{DC})$$

If discharge signal is activated, the coupling between the models vanishes and only the term:

$$f_{DC} = -b_{TP2}\dot{m}_{TPin,nom}d_{DC}$$

remains active. The resulting DC line model introduced in the Eq. 3.9 employs the blender inlet mass flow and TP turret speed as input-, and TP fill level as the output signal. As all input signals can be manipulated, and the output signal can be measured, this representation can be employed for the development of TP fill level control concept in the further course of the thesis.

3.3 Modeling of HME Line

HME line consists of three individual modeling units, namely, extrusion concentration, extrusion mass flow, and cooling & pelletisation modeling unit. Table 3.2 provides a compact overview of the individual modeling units and the associated input- and output variables.

Table 3.2: Overview of the individual modeling units in the HME line

Modeling Unit	**Input Signals**	**Output Signals**
Extrusion: Concentration	inlet concentration c_{HMEin}	outlet concentration c_{HMEout}
Extrusion: Mass flow	inlet mass flow $\dot{m}_{Ein}$	outlet mass flow $\dot{m}_{Eout}$
Cooling & Pelletisation	pelletizer intake speed air pressure extrusion die temperature	strand temperature strand diameter

3.3.1 Modeling of Extrusion Concentration

Transfer function relating the concentrations at the extruder inlet and the extruder outlet can be identified in the exactly same manner as introduced in the Subsection 3.2.1. Extrusion inlet concentration corresponds to:

$$c_{HMEin} = \frac{\dot{m}_{API}}{\dot{m}_{API} + \dot{m}_{POL}}$$

and can be adjusted by manipulating the ratio of API and polymer feeder mass flows. Extrusion outlet concentration can be measured by means of the NIR spectrometer placed on the adapter plate at the extruder outlet. The experimental data required for the identification purposes is collected via laboratory experiment involving the gradual changes of inlet concentration. The collected input- and output sequences are imported to the System Identification Toolbox App. Again, a

transfer function structure proposed in the [Kruisz et al., 2017] is chosen. Outcome of the identification procedure is a proper transfer function third order, from now on referred to as HME concentration transfer function:

$$P_{C,HME}(s) = \frac{C_{HMEout}(s)}{C_{HMEin}(s)} = e^{-sT_{d,HME}} P^{\star}_{HME}(s)$$

$$\text{with} \qquad P^{\star}_{HME}(s) = \frac{1.93 \cdot 10^{-5}}{s^3 + 0.1253s^2 + 0.00245s + 1.93 \cdot 10^{-5}} \qquad \text{and} \qquad T_{d,HME} = 62.5s$$

3.3.2 Modeling of Extrusion Mass Flow

This Subsection outlines the identification of the state-space-model relating the extrusion inlet- and outlet mass flow. The identification procedure follows in the Subsection 3.2.2 introduced steps.
Extrusion inlet mass flow corresponds to the sum of API and polymer feeder mass flows, i.e. $\dot{m}_{Ein} = \dot{m}_{API} + \dot{m}_{POL}$.
Extrusion outlet mass flow is indirectly measured via catch scale, which is brought into the device setup and placed at the extruder outlet.

$$\dot{m}_{Eout}(t) = \frac{dm_{Eout}(t)}{dt} \approx \frac{m_{Eout}(t) - m_{Eout}(t - \Delta t)}{\Delta t} \qquad \text{with} \quad \Delta t = T_s$$

Due to the high frequency components, in such a way evaluated outlet mass flow needs to be filtered with low-pass zero-phase forward-backward filter. The best filtering results are obtained with the filter order 62 and cut-off frequency $f_{cut} = 0.025Hz$. Again, the catch-scale is not a permanent part of the device setup, and the extrusion outlet mass flow cannot be measured in the continuous plant operation.
Experimental data required for the identification purposes is collected via in the Fig. 3.6 depicted laboratory experiment. Extrusion inlet mass flow is varied around the nominal point in $\pm 0,25kg/h$ and $\pm 0,5kg/h$ steps. Extrusion inlet mass flow, i.e. its deviation from the nominal operating point, is imported as the input sequence, and the extrusion outlet mass flow, i.e. its deviation from the nominal operating point, as the output sequence to the System Identification Toolbox. Model order is set to two, and the estimation horizons are appropriately adapted. Remaining specifications and estimation options correspond to the ones suggested in the MATLAB code introduced in the Subsection 3.2.2. Outcome of the identification procedure is a time-discrete state-space-model, from now on referred to as extrusion model, which reads as:

$$\begin{aligned} x_{E,k+1} &= A_E x_{E,k} + b_E \Delta u_{E,k} \\ \Delta y_{E,k} &= c_E^T x_{E,k} + d_E \Delta u_{E,k} \end{aligned} \tag{3.10}$$

with: $u_E = \dot{m}_{Ein}$ and $\Delta u_E = u_E - u_{E,nom}$ and $y_E = \dot{m}_{Eout}$ and $\Delta y_E = y_E - y_{E,nom}$
Nominal operating points of extrusion inlet- and outlet mass flow are $\dot{m}_{Ein} = \dot{m}_{Eout} = 2kg/h$. Fig. 3.6 illustrates the process variables of interest during the executed laboratory experiment. The model estimated outlet mass flow corresponds to the measured outlet mass flow, which verifies the quality of the identified extrusion model.

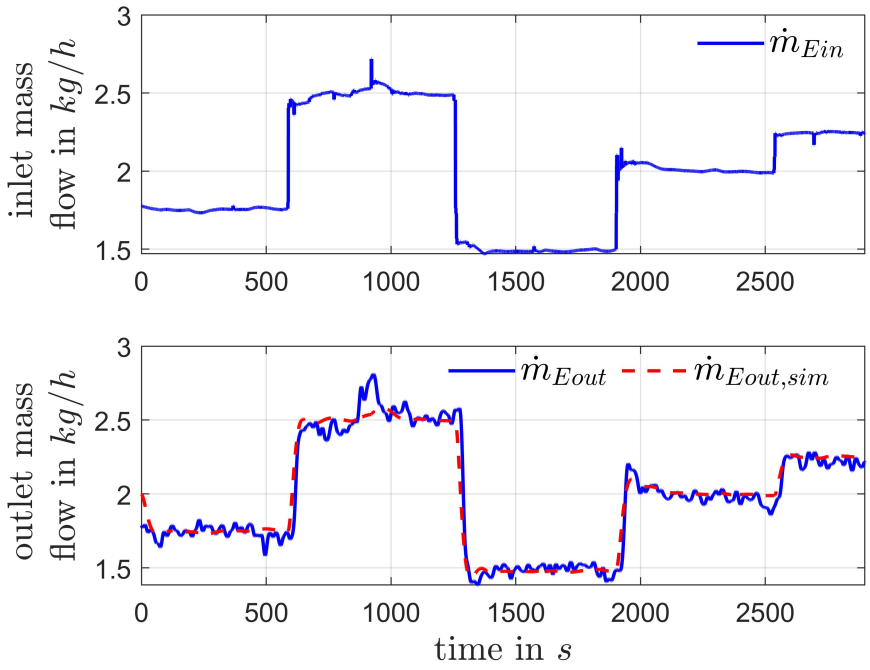


Figure 3.6: Laboratory Experiment-Identification of extrusion mass flow model: First subplot depicts the set extrusion inlet mass flow. Second subplot depicts the measured- and via identified model estimated extrusion outlet mass flow in blue and red color, respectively.

3.3.3 Modeling of Cooling & Pelletisation Unit

Model identification of cooling & pelletisation unit tends to be more complicated compared to other modeling units. This unit exhibits the nonlinear characteristics and cannot be modeled via previously introduced linear data-driven identification approaches. Therefore, a local-linear-model-tree (LoLiMoT) algorithm for data-driven identification of nonlinear systems, suggested in [Nelles et al., 2000], is utilized. Again, the experimental data is collected via appropriate laboratory experiments, pre-processed and provided to the LoLiMoT algorithm. Outcome of the LoLiMoT identification procedure are two Neuro-Fuzzy-Models (NFM), one NMF for each model output. Obtained NFMs relate the strand temperature and strand diameter as output- and pelletizer intake speed, air pressure and extrusion die temperature as the input variables. [Rehrl et al., 2019] develops a control strategy combining LoLiMoT and MPC for improving the pellet quality. However, the developed model and control concept do not have an impact on the process quantities of interest within the scope of the thesis. Therefore, this model is replaced with simple time-delay element $e^{-sT_{d,PC}}$. Time duration of cooling & pelletisation unit is measured via stopwatch $T_{d,PC} = 9s$ and can be utilized for the simulation purposes.

3.3.4 Modeling of Feeding

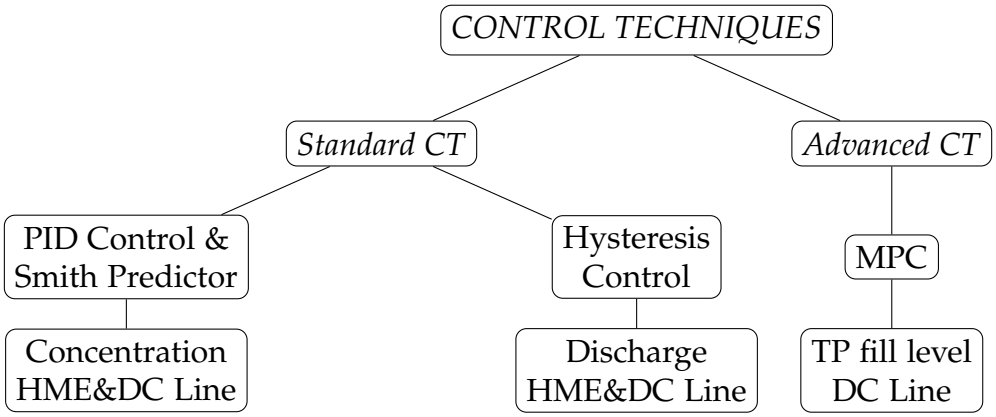
Transfer function relating the feeder mass flow set point as the input-, and the real feeder mass flow as the output variable, reads as:

$$P_{FEED}(s) = \frac{100}{s + 100}$$

This transfer function describes the dynamic of all introduced feeders, i.e. API, polymer, pellet and pre-mix feeder. As this dynamic is very fast compared to the dynamics of other modeling units, it is employed for plant modeling in the simulation, yet neglected by the controller design.

4 Control Design & Implementation

This Chapter contains a detailed description of a control design and its implementation.
A theoretical background of the standard PID controller is introduced in the Section 4.1. Furthermore, the relevant topics, such as the discretization of time-continuous control laws and anti-windup strategy are addressed within the Subsections 4.1.1 and 4.1.2, respectively. Section 4.2 provides a theoretical background on the control of time-delay systems, i.e. Smith predictor structure. MPC functionality and implementation, as well the observer design for the systems with no- or only partially measurable states, are addressed within the Section 4.5.
Sections 4.3, 4.4 and 4.6 employ the introduced control techniques for a development of the individual control concepts, i.e. HME & DC line concentration controller, HME & DC line discharge controller and TP fill level controller, respectively.
Individually developed control concepts are joined to a DC line controller in the Section 4.7. DC line controller is investigated, not only in the simulation, but on the real system, as well. In order examine the controller on the real system, the proposed control laws need to be adapted for the execution on a XAMControl Automation software platform. MATLAB Compiler is utilized to create a dynamic-link library (*dll*) from the MATLAB code, i.e. provided *.m* functions are joined to a library by means of the code generation. In order to employ the created *dll*, a MATLAB Runtime needs to be installed on a target system with XAMControl Automation Software platform. Via this approach, the exactly same control algorithms are applied in the simulation, and on the real plant. The control laws are executed with a sampling time of $T_s = 0.35s$. The required control inputs are gathered from the process equipment/sensors, and the calculated control outputs are fed to the process equipment/actuators, both via XAMControl software. A final controller tuning is performed on the real system.
A following tree structure allows to distinguish between the introduced control techniques and their application domains in the continuous production plant.



4.1 Proportional-Integral-Derivative (PID) Control: Theoretical Background

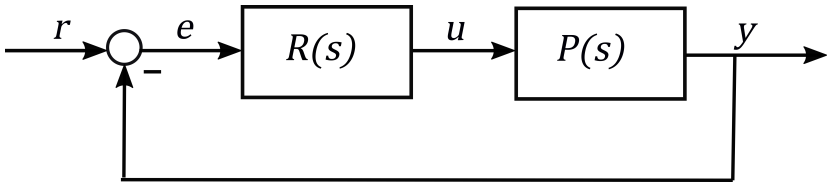


Figure 4.1: Standard unity-feedback loop: $R(s)$ denotes controller- and $P(s)$ denotes plant/system transfer function. Signal notation: r reference signal, e control error, u actuating (plant input) signal, y controlled (plant output) signal

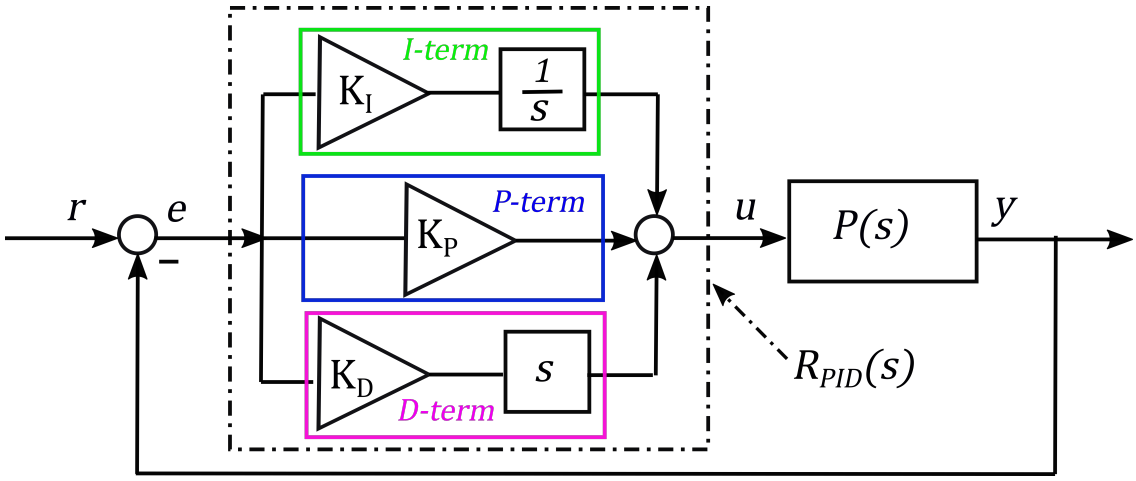


Figure 4.2: Implementation of PID controller within the standard unity-feedback loop: $R_{PID}(s)$ consists of proportional (P), integral (I) and derivative (D) term.

A proportional-integral-derivative (*PID*) control is a well established, standard control technique, widely used in the different industry domains. Due to its effectiveness and straightforward and simple design, it is an intuitive first choice of control algorithm for diverse applications. Using in the Eq. 4.1 introduced control law, PID controller manipulates the plant input u, and drives the plant output y to

the reference signal r and keeps it there. A difference between the system output and reference signal, denoted as a control error $e = r - y$, is utilized as the input signal for the PID controller.
A time-domain equation, relating the control error $e(t)$ and the plant input $u(t)$ is given in a following form:

$$u(t) = \underbrace{\underbrace{K_P e(t)}_{\text{P-term}} + \underbrace{K_I \int_0^t e(\tau)d\tau}_{\text{I-term}} + \underbrace{K_D \frac{de(t)}{d(t)}}_{\text{D-Term}}}_{\text{PID control law}} \tag{4.1}$$

In such a way calculated signal $u(t)$ consists of following three terms:

1. proportional (P-) term weighting the current value of control error.
2. integral (I-) term weighting the integral over the control error. Integral computation is executed over all previous values of the control error.
3. derivative (D-) term weighting the control error time-derivative, i.e. the current rate of the control error.

Note: The parameters K_P, K_I and K_D are real and constant terms weighting the control error, integral over it, and its derivative, respectively. Incorporation of $K_I := \frac{K_P}{T_I}$ and $K_D := K_P T_D$ in the Eq. 4.1 leads to an alternative PID formulation:

$$u(t) = K_P[e(t) + \frac{1}{T_I}\int_0^t e(\tau)d\tau + T_D \frac{de(t)}{d(t)}] \tag{4.2}$$

This representation is beneficial, as the parameters T_I and T_D have a physical meaning, denoting the real integration- and derivative time. Proportional gain K_P can be denoted as the gain of the system.
Laplace transformation of a time-domain representation introduced in the Eq. 4.1 leads to a formulation of the PID transfer function in the s-domain.
Note: This relation holds under the assumption, that the initial value of control error equals zero.

$$\bar{u}(s) = K_P\bar{e}(s) + \frac{K_I}{s}\bar{e}(s) + sK_D\bar{e}(s) = K_P[\bar{e}(s) + \bar{e}(s)\frac{1}{sT_I} + sT_D\bar{e}(s)]$$

The resulting transfer function $R_{PID}(s)$ relates the Laplace transforms $\bar{u}(s)$ and $\bar{e}(s)$ of PID output and input signal, respectively.

$$R_{PID}(s) = \frac{\bar{u}(s)}{\bar{e}(s)} = K_P + \frac{K_I}{s} + K_D s = K_P(1 + \frac{1}{sT_I} + sT_D)$$

The last introduced term sT_D indicates the improperness of the derived transfer function (*a nominator degree is higher than a denominator degree*). This is a consequence of the incorporated derivative term, which makes this transfer function unsuitable for any practical implementation. Therefore, in the practical approaches, this term is substituted by the shifted derivative term $\frac{s}{1+sT}$, which leads to a proper transfer function:

$$\begin{aligned} R_{PID}(s) &= K_P + \frac{K_I}{s} + K_D\frac{s}{1+sT} \\ R_{PID}(s) &= R_{PI}(s) + R_D(s) \end{aligned} \tag{4.3}$$

From this point on, this transfer function is going to be referred to as a time-continuous PID controller transfer function.

4.1.1 Discretization of Time-Continuous Control Law

In order to implement the PID controller on the real system with the specified sampling time T_s, the time-continuous control laws introduced in Eq. 4.1 and 4.3 need to be discretized.

Discretization of the Proportional-Integral Part

PI controller in time-domain

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau)d\tau$$

PI controller transfer function

$$R_{PI}(s) = K_P + \frac{K_I}{s}$$

Control law is rewritten in the discrete time points, with index k denoting a signal value at the discrete time instant (iteration) k, i.e. $u_k = u(kT_s)$ and $e_k = e(kT_s)$:

$$u_k = K_P e_k + K_I \int_0^{kT_s} e(\tau)d\tau$$

$$u_k = K_P e_k + K_I \int_0^{(k-1)T_s} e(\tau)d\tau + \int_{(k-1)T_s}^{kT_s} e(\tau)d\tau + K_P e_{k-1} - K_P e_{k-1}$$

with $u_{k-1} = K_P e_{k-1} + K_I \int_0^{(k-1)T_s} e(\tau)d\tau$

$$u_k = K_P e_k + \underbrace{K_P e_{k-1} + K_I \int_0^{(k-1)T_s} e(\tau)d\tau}_{u_{k-1}} - K_P e_{k-1} + K_I \int_{(k-1)T_s}^{kT_d} e(\tau)d\tau$$

$$u_k = u_{k-1} + K_P e_k - K_P e_{k-1} + \underbrace{K_I \int_{(k-1)T_s}^{kT_s} e(\tau)d\tau}_{\text{Approximation term - A}}$$

The last integral term A, i.e. the surface which control error function $e(t)$ closes with the time axis t, can be approximated by means of the different discretization methods:

Inward Euler: $A = e_{k-1}T_s$

Backward Euler: $A = e_k T_s$

Tustin: $A = \frac{1}{2}(e_k + e_{k-1})T_s$

Inward Euler discretization method leads to the following formulation of the final difference equation as:

$$u_k = u_{k-1} + K_P e_k + (K_I T_s - K_P)e_{k-1} \tag{4.4}$$

$$\tilde{u}(z) - \tilde{u}(z)\frac{1}{z} = K_P\tilde{e}(z) + \frac{1}{z}(-K_P + K_I T_s)\tilde{e}(z) \tag{4.5}$$

Z-transformation of the difference equation introduced in the Eq. 4.4 leads to a formulation of the time-discrete PI controller transfer function.
Note: This relation holds under the assumption, that the initial value of control

error equals zero.

$$\begin{aligned} R_{PI}(z) = \frac{\tilde{u}(z)}{\tilde{e}(z)} &= \frac{K_P(z-1) + K_I T_s}{z-1} \\ &= K_P + \frac{K_I T_s}{z-1} \\ &= R(s)|_{s=\frac{z-1}{T_s}} \end{aligned} \tag{4.6}$$

Resulting time-discrete transfer function indicates that only the integral term undergoes the discretization, while the proportional term stays unaffected.

Discretization of the Derivative Component

In the Eq. 4.6 derived s to z-domain mapping relation is employed for the discretization of the derivative component as:

$$\begin{aligned} R_D(s) &= K_D \frac{s}{1 + sT_s} \\ R_D(z) &= R_D(s)|_{s=\frac{z-1}{T_s}} \\ &= \frac{K_D}{T_s} \frac{z-1}{z} \end{aligned}$$

This leads to the final formulation of the time-discrete PID controller transfer function:

$$R_{PID}(z) = \frac{\tilde{u}(z)}{\tilde{e}(z)} = \frac{(K_P + \frac{K_D}{T_s}) + (K_I T_s - K_P - 2\frac{K_D}{T_s})z^{-1} + \frac{K_D}{T_s}}{1 - z^{-1}} \tag{4.7}$$

This formulation allows the straightforward development of the difference equations for the application of the PID algorithm on the real system:

$$u_k = u_{k-1} + (P + D) \cdot e_k + (I - P - 2D) \cdot e_{k-1} + D \cdot e_{k-2}$$

$$\text{with the coefficients:} \qquad P = Kp, \quad I = K_I T_s \quad \text{and} \quad D = \frac{K_d}{T_s}$$

4.1.2 Integrator Windup & Anti-Windup Strategy

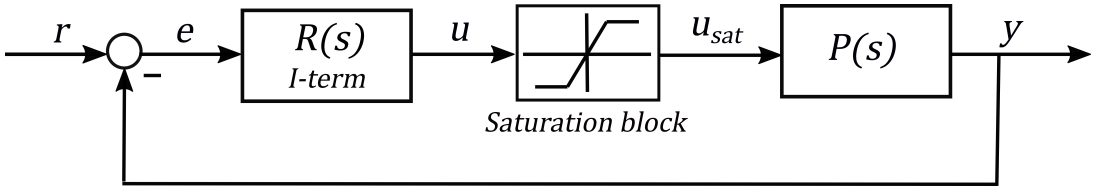


Figure 4.3: Integrator Windup: Standard unity-feedback loop involving an integral controller and saturation element

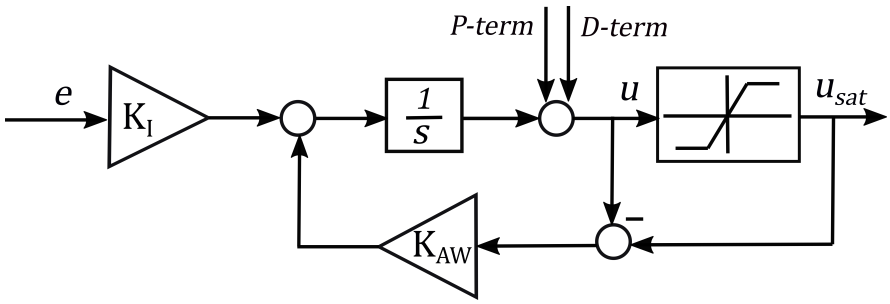


Figure 4.4: Implementation of anti-windup strategy for a PID controller

Although very effective in many applications, implementation of the standard PID controller can lead to the undesired process events, such as the integrator windup effect. Integrator windup occurs in the systems, which involve a combination of an integral controller and a saturation element, as depicted in the Fig. 4.3. Saturation element realizes the following relation:

$$u_{sat}(t) = \begin{cases} u_{min}, & \forall\ u(t) < u_{min} \\ u_{max}, & \forall\ u(t) > u_{max} \\ u(t), & otherwise \end{cases}$$

The windup effect occurs, as the actuating variable $u(t)$ goes into saturation. Although the actuating variable reaches the allowed peak value u_{min} or u_{max}, implemented controller continues to integrate. Due to this permanent integration, controller continues to increase or decrease the actuating variable, leading to the integrator windup effect. In order to prevent this undesired behavior, one needs to find a way to 'stop' the integration if the actuating variable enters the saturation. An effective anti-windup strategy implies a modification of the controller structure, as suggested in the Fig. 4.4. The difference between the calculated- and applied controller output $u - u_{sat}$ is weighted with the anti-windup gain K_{AW} and integrated.
PID control s-domain relation can be extended in the following manner:

$$\begin{aligned} \tilde{u}(s) &= K_P\tilde{e}(s) + \frac{K_I}{s}\tilde{e}(s) + K_D\frac{s}{1+sT_s}\tilde{e}(s) + \frac{K_{AW}}{s}\tilde{e}_u(s) \\ \bar{u}(z) &= K_P\bar{e}(z) + \frac{K_I T_s}{z-1}\bar{e}(z) + \frac{K_D}{T_s}\frac{z-1}{z}\bar{e}(z) + \frac{K_{AW}T_s}{z-1}\bar{e}_u(z) \\ \text{with} \quad & \tilde{e}_u(s) = \tilde{u}_{sat}(s) - \tilde{u}(s) \quad \text{and} \quad \bar{e}_u(z) = \bar{u}_{sat}(z) - \bar{u}(z) \end{aligned} \tag{4.8}$$

The time-discrete relation introduced in the Eq. 4.8 can be transformed to a difference equation and utilized for the application of the controller algorithm on

the real system in the straightforward manner. Implementation of an anti-windup strategy significantly increases the performance of PID controller.

4.2 Smith Predictor: Theoretical Background

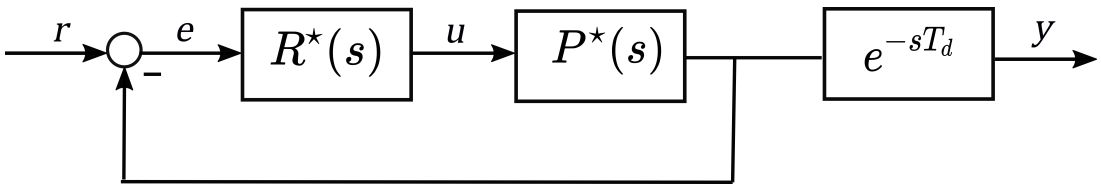


Figure 4.5: Closed feedback loop with delay-time transfer function

In order to control the systems with a non neglectable time-delay, standard control techniques are usually combined with the Smith predictor structure.
Plant transfer function is given as:

$$P(s) = P^\star(s)e^{-sT_d}$$

with the time delay free transfer function $P^\star(s)$ and time delay element e^{-sT_d}. Controller transfer function R_s should be developed in such a way, that in the Fig. 4.5 depicted feedback loop results in the same overall transfer function $T_d(s)$, as the standard unity feedback loop whose structure is depicted in the Fig. 4.1. Overall transfer functions are defined as:

$$T(s) = \frac{P(s)R(s)}{1 + P(s)R(s)} \qquad \text{Fig. 4.1}$$

$$T_d(s) = \frac{P^\star(s)R^\star(s)e^{-sT_d}}{1 + P^\star(s)R^\star(s)} \qquad \text{Fig. 4.5}$$

In that sense, Smith predictor controller transfer function computes as:

$$R(s) = \frac{R^\star(s)}{1 + R^\star(s)P^\star(s)(1 - e^{-sT_d})} \tag{4.9}$$

Fig. 4.6 depicts a standard unity feedback loop structure, which involves in the Eq. 4.9 introduced controller transfer function.

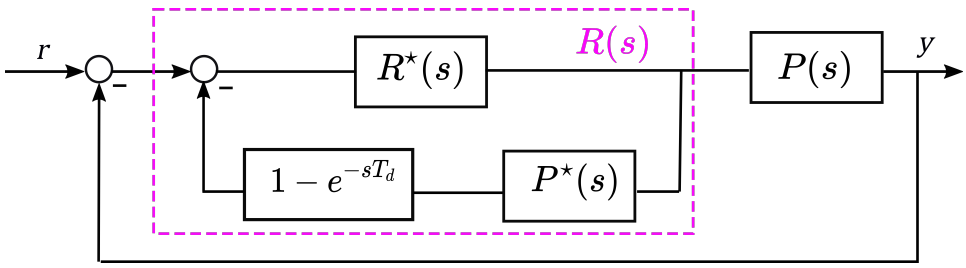


Figure 4.6: Initial Smith predictor structure

In the Fig. 4.7 depicted feedback loop structure is equivalent to the structure presented in the Fig. 4.6. Introduced modifications are solely structural, the functionality remains unaffected. Real system application would require a transition of the developed Smith predictor structure from the time-continuous to the time-discrete domain. Continuous transfer functions, i.e. $R^\star(s)$, $P^\star(s)$ and $P(s)$, should be substituted with the corresponding discrete transfer functions, i.e. $R^\star(z)$, $P^\star(z)$ and $P(z)$. Time-delay element can be substituted via following relation:

$$e^{-sT_d} \quad \longrightarrow \quad z^{-n_d}$$
$$\text{with} \quad n_d = \frac{T_d}{T_s}$$

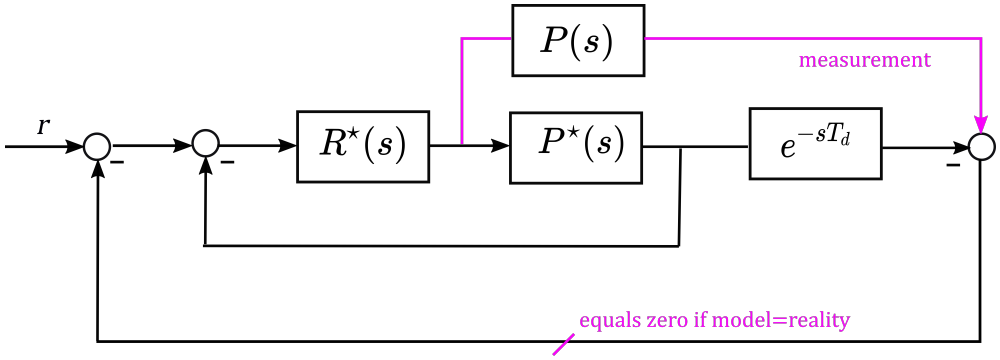


Figure 4.7: Final Smith predictor structure

Table 4.1: Overview of concentration controller parameters.

Controller	P	I	D	K_{AW}	n_d
HME	1.5	0.01	0.7	1	186
DC	3.5	0.1	0.175	1	20

4.3 Concentration Controller

First control objective in the continuous production plant is to keep the API concentration within the specified range around the reference concentration. Therefore, a concentration controller needs to be designed and implemented.
This controller can be employed for the feedback control of extrusion- or blender outlet concentration in the HME or DC line, respectively. Concentration controller employs by means of the NIR measured concentration as the measured process variable.
Both, HME and DC concentration transfer functions (see Chapter 2) exhibit a certain time delay. Therefore, a Smith predictor structure depicted in the Fig. 4.7 is utilized. Control of the time delay free dynamic takes places via in the Section 4.1 introduced PID controller. Inlet concentration is always bounded between 0 and 1. Therefore, in the Subsection 4.1.2 introduced anti-windup strategy is utilized.
Concentration controller adjusts the blender or extrusion inlet concentration. As this quantity cannot be directly manipulated, this is realized via manipulation of API and polymer feeder mass flow ratio in the HME, or of the pellet and pre-mix feeder mass flow ratio in the DC line. Concentration controller ensures the capability of the plant to react and compensate brief feeder disturbances or material impurities. Fig. 4.8 depicts a final structure of the concentration controller. Control parameters are tuned in the simulation. Tab. 4.1 provides an overview of final control parameter, for DC and HME concentration controller.

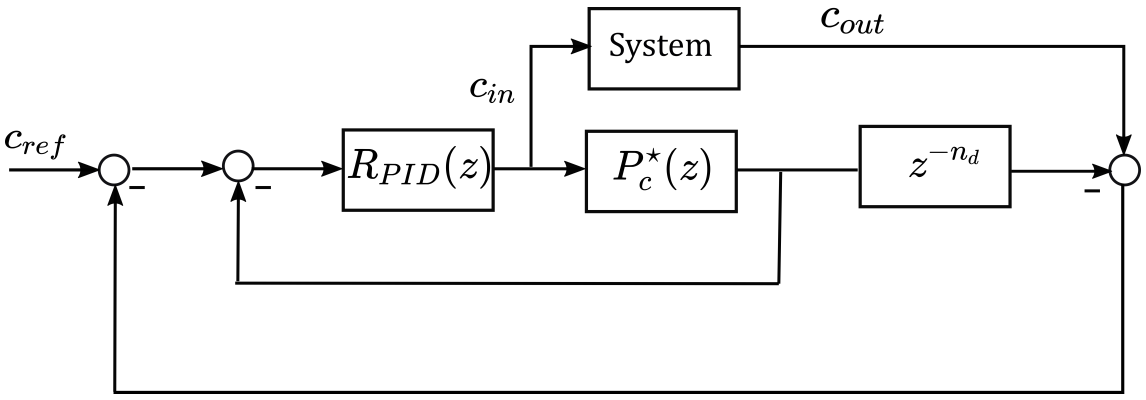


Figure 4.8: Schematic representation of concentration controller

4.4 Discharge Controller

If the concentration controller does not accomplish to keep the API concentration within the allowed range, material needs to be discharged. Both, HME and DC line involve one discharge unit. In the Fig. 4.9 depicted controller is employed for the discharge (de)activation in the both lines. Discharge controller utilizes the NIR measured blender- and extrusion outlet concentration in the DC and HME line, as the measured process variable. Concentration error $c_e = c_{out} - c_{th}$ is evaluated and employed as the function argument for the hysteresis controller depicted in the Fig. 4.10. Allowed concentration range changes depending on the discharge signal. If discharge is not activated, the controller follows the red line with the bounds $[-c_{th}, c_{th}]$. Otherwise, controller follows the blue line with the tighter bounds, i.e. $[-c_{th} + h, c_{th} - h]$. Control parameters $c_{th} = 0.02$ and $h = 0.01$ stay the same in the both lines.
Discharge flap in the DC line is placed at the blender outlet and discharge is triggered by the current NIR measurement. On the other hand, NIR measurement in the HME line takes place at the extruder outlet, while the discharge takes place at the pellet hopper. Therefore, a time-delay of cooling & pelletisation unit needs to be considered. If the measured concentration does not lie in the allowed range, discharge signal is activated after six seconds. On the other hand, if discharge is activated, and measured concentration returns to the allowed range, discharge is deactivated after nine seconds.

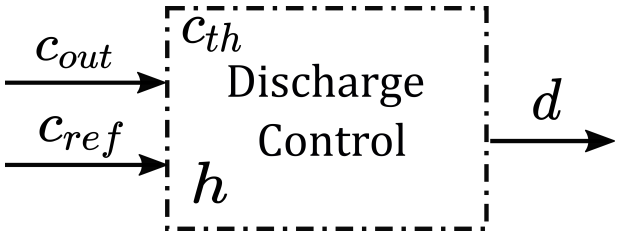


Figure 4.9: Schematic representation of discharge controller: Controller with two adjustable parameters c_{th} and h is triggered by the measured outlet concentration.

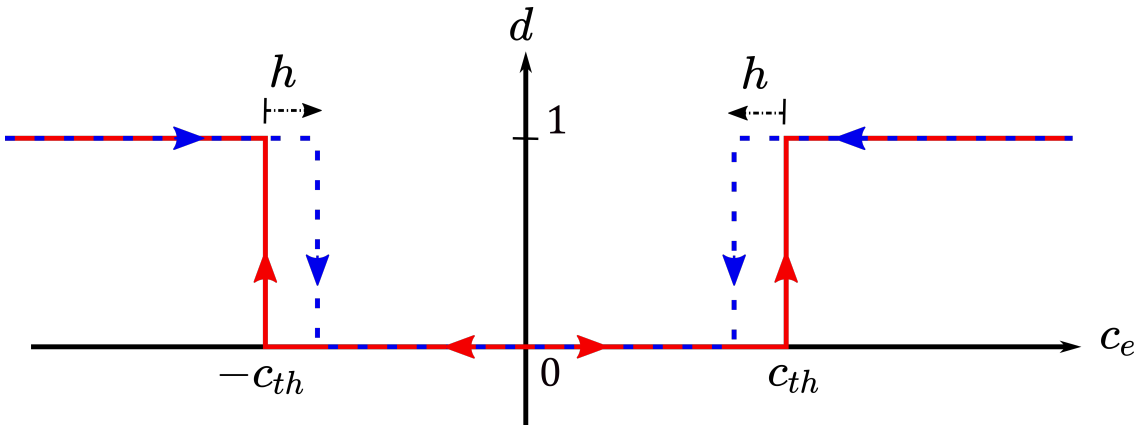


Figure 4.10: Characteristic curve of discharge controller

4.5 Model Predictive Control: Theoretical Background

Basic idea[1]*:*
An MPC operates like a good chess player: It plans a certain number of future moves in an optimal sense, according to the current conditions and the predicted future developments. Than, it executes first one of them, waits on the opponents next move (system response), and repeats the same procedure.

MPC application yields many benefits compared to the standard control techniques. One of the most significant features is a relatively simple consideration of input-, input rate-, output- and state constraints, i.e. the possibility of its reduction to the input rate constraints solely. Another important feature is a straightforward transfer between SISO and MIMO systems. In addition to these, MPC implementation does not invoke integrator windup, typical for the previously introduced PID controller. Furthermore, it is particularly suitable for controlling the systems of high complexity.

Requirements for MPC design:

- Identified time-discrete model of a system given in following form:

$$x_{k+1} = f(x_k, u_k)$$
$$y_k = g(x_k, u_k)$$

  Note 1: Index k denotes a signal value at the discrete time instant (iteration) k, e.g. $x_k = x(kT_s)$.
  Note 2: Model is given in the general notation, as the MPC can be designed for both, linear and nonlinear systems.
- Input signal from the last time instant: u_{k-1}
- State vector at the current time instant: x_k
- Reference value for the output signal y_k at the current time instant: r_k

The MPC procedure can be summarized in a following manner: Dependent on the system of interest, a current state vector x_k is either directly measured, or estimated by an observer. Using the known signals x_k and u_{k-1}, as well as the system describing quantities (parameters of the functions f and g), MPC predicts future evolution of the signals of interest over the prediction horizon n_p. The control objectives are defined by means of an objective function J. A suitable objective function is formulated and rewritten as the function of the known quantities via recursive development of system equations. Input signal- or input signal rate sequence over the control horizon n_c is introduced as the optimization variable. MPC solves an optimization problem using the state-of-the-art numerical algorithm and forwards the calculated input signal to a system. Finally, the next state vector x_{k+1} is obtained and the introduced procedure reiterated.

Despite the fact, that the optimal input sequence is calculated over the complete control horizon $(\hat{u}_k, \hat{u}_{k+1}, \ldots, \hat{u}_{k+n_c-1})$, only the first element of this sequence $(\hat{u}_k)$

[1] *Automatisierung mechatronischer Systeme*, Lecture transcript

is applied as the actuating signal to the system. Optimization is than re-executed and optimization interval shifted by one time step (*receding horizon*). A number of optimization variables in each iteration corresponds to the control horizon n_c. Therefore, n_c needs to be kept as low as possible, yet without degradation of control the controller functionality, i.e. this value should be chosen as a compromise between the control efficiency and complexity.

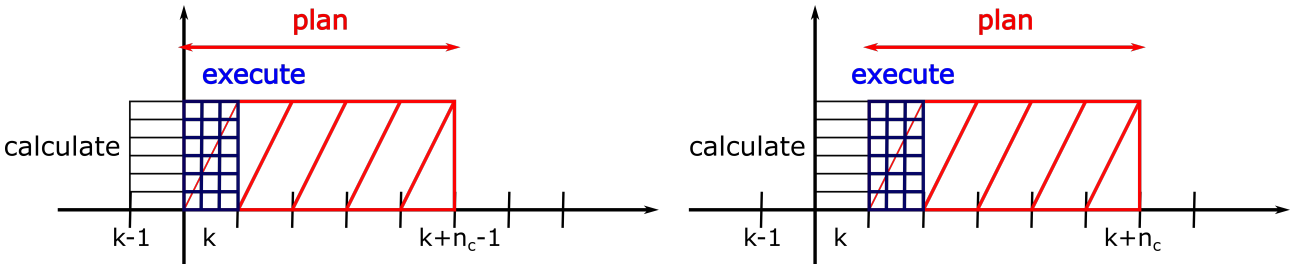


Figure 4.11: MPC agenda: MPC plans n_c future moves and executes the first one of them. Optimization interval (red) is than shifted by one time step (*receding horizon*).

4.5.1 Recursive Development of System Equations:

In general, it can be distinguished between the linear and nonlinear MPC. Within the scope of this thesis, a focus is put on the linear MPC design. From this point on, following linear, time invariant, discrete system is considered as the MPC starting point:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + B_d d_k \\ y_k &= Cx_k \end{aligned} \tag{4.10}$$

with: $x \in R^n$, $u \in R^m$, $d \in R^g$, $y \in R^p$, i.e. MIMO system with m inputs, g disturbances, n states and p outputs, and system describing parameters $A \in R^{nxn}$, $B \in R^{nxm}$, $B_d \in R^{nxg}$, $C \in R^{pxn}$.
In the Section 3 performed process modeling provides the information, that the system of interest is strictly proper. Therefore, u_k and d_k terms are omitted within the Eq. 4.10. Yet, the presented approach can be extended to a bi-proper case in simple and straightforward manner.
A disturbance d_k is assumed to be known and constant over a prediction horizon.

Idea: One can present the input signal in the relation to the step before, i.e. $u_k = u_{k-1} + \Delta u_k$. By inserting this relation to Eq. 4.10, one obtains a recursive formulation of difference equations.

Prediction of the state x over the prediction horizon n_p:

$$\begin{aligned}
\hat{x}_{k+1} &= Ax_k + Bu_{k-1} + B\Delta u_k + B_d d_k \\
\hat{x}_{k+2} &= A^2 x_k + (AB+B)u_{k-1} + (AB+B)\Delta u_k + B\Delta u_{k+1} + (AB_d + B_d)\underbrace{d_{k+1}}_{=d_k} \\
&\vdots \\
\hat{x}_{k+n_c} &= A^{n_c} x_k + (A^{n_c-1} + \ldots + A + E)(Bu_{k-1} + B_d d_k) + \sum_{j=1}^{n_c}(A^{n_c-j} + \ldots + A + E)B\Delta\hat{u}_{k+j-1}
\end{aligned}$$

The input signal u stays constant after the control horizon: $\Delta\hat{u}_{k+j} = 0 \;\; \forall j \geq n_c$.

$$\begin{aligned}
\hat{x}_{k+n_c+1} &= A^{n_c+1} x_k + (A^{n_c} + \ldots + A + E)(Bu_{k-1} + B_d d_k) + \sum_{j=1}^{n_c}(A^{n_c+1-j} + \ldots + A + E)B\Delta\hat{u}_{k+j-1} \\
&\vdots \\
\hat{x}_{k+n_p} &= A^{n_p} x_k + (A^{n_p-1} + \ldots + A + E)(Bu_{k-1} + B_d d_k) + \sum_{j=1}^{n_c}(A^{n_p-j} + \ldots + A + E)B\Delta\hat{u}_{k+j-1}
\end{aligned}$$

Prediction of the output y over the prediction horizon n_p:

$$\begin{aligned}
\hat{y}_{k+1} &= C\hat{x}_{k+1} \\
\hat{y}_{k+2} &= C\hat{x}_{k+2} \\
&\vdots \\
\hat{y}_{k+n_p} &= C\hat{x}_{k+n_p}
\end{aligned}$$

Compact representation of the introduced quantities via vector notation

Algebraic vectors $\bar{x}_{k+1}$ and $\bar{y}_{k+1}$ hold the predicted state- and output signals over the prediction horizon, respectively:

$$\bar{x}_{k+1} = \begin{bmatrix} \hat{x}_{k+1} \\ \hat{x}_{k+2} \\ \vdots \\ \hat{x}_{k+n_p} \end{bmatrix}, \;\; \bar{x}_{k+1} \in R^{n \cdot n_p x 1} \qquad \bar{y}_{k+1} = \begin{bmatrix} \hat{y}_{k+1} \\ \hat{y}_{k+2} \\ \vdots \\ \hat{y}_{k+n_p} \end{bmatrix} = \begin{bmatrix} C\hat{x}_{k+1} \\ C\hat{x}_{k+2} \\ \vdots \\ C\hat{x}_{k+n_p} \end{bmatrix}, \;\; \bar{y}_{k+1} \in R^{p \cdot n_p x 1} \tag{4.11}$$

Algebraic vectors $\bar{u}_k$ and $\Delta\bar{u}_k$ hold the calculated input- and input rate signals over

the control horizon, respectively:

$$\bar{u}_k = \begin{bmatrix} \hat{u}_k \\ \hat{u}_{k+1} \\ \vdots \\ \hat{u}_{k+n_c-1} \end{bmatrix}, \quad \bar{u}_k \in R^{m \cdot n_c x 1} \qquad \Delta\bar{u}_k = \begin{bmatrix} \Delta\hat{u}_k \\ \Delta\hat{u}_{k+1} \\ \vdots \\ \Delta\hat{u}_{k+n_c-1} \end{bmatrix}, \quad \Delta\bar{u}_k \in R^{m \cdot n_c x 1} \tag{4.12}$$

$$\bar{u}_k = L \underbrace{u_{k-1}}_{\text{known quantity}} + M \overbrace{\Delta\bar{u}_k}^{\text{optimization variable}}$$

$$\text{, with} \quad L = \begin{bmatrix} I \\ I \\ \vdots \\ I \end{bmatrix}, \quad L \in R^{m \cdot n_c x m}, \quad M = \begin{bmatrix} I & 0 & \dots & \dots & 0 \\ I & I & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \\ I & \dots & \dots & \dots & I \end{bmatrix}, \quad M \in R^{m \cdot n_c x m \cdot n_c}$$

and the identity matrix $I \in R^{mxm}$

Final prediction formulation

$$\bar{y}_{k+1} = \underbrace{Fx_k + Gu_{k-1} + G_d d_k}_{\text{known quantites}} + H \overbrace{\Delta\bar{u}_k}^{\text{optimization variable}} \tag{4.13}$$

with:

$$F = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{n_p} \end{bmatrix}, \quad G = \begin{bmatrix} CB \\ C(A+E)B \\ \vdots \\ C(A^{n_p-1} + \dots + A + E)B \end{bmatrix}, \quad G_d = \begin{bmatrix} CB_d \\ C(A+E)B_d \\ \vdots \\ C(A^{n_p-1} + \dots + A + E)B_d \end{bmatrix}$$

$$H = \begin{bmatrix} CB & 0 & \dots & 0 \\ C(A+E)B & CB & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ C(A^{n_c-1}+\dots+A+E)B & C(A^{n_c-2}+\dots+A+E)B & \dots & CB \\ C(A^{n_c}+\dots+A+E)B & C(A^{n_c-1}+\dots+A+E)B & \dots & C(A+E)B \\ \vdots & \vdots & \ddots & \vdots \\ C(A^{n_p-1}+\dots+A+E)B & C(A^{n_p-2}+\dots+A+E)B & \dots & C(A^{n_p-n_c}+\dots+A+E)B \end{bmatrix}$$

, with $F \in R^{p \cdot n_p x n}$, $G \in R^{p \cdot n_p x m}$, $G_d \in R^{p \cdot n_p x g}$ and $H \in R^{p \cdot n_p x m \cdot n_c}$

4.5.2 Formulation of Objective Function

A choice of the objective function J is an essential step by the MPC design. The parameters and structure of implemented objective function have a direct influence on the effectiveness of the proposed control strategy.
In the Eq. 4.14 introduced objective function can be interpreted as follows: First term penalizes the difference between the predicted output and the reference signal, i.e. the predicted control error $\hat{e}_{k+i}$, over the prediction horizon. The second and third term penalize the magnitude and the rate of the input signal, respectively.

$$J = \underbrace{\sum_{i=1}^{n_p} \underbrace{(\hat{y}_{k+i} - r_{k+i})^T}_{\hat{e}_{k+i}^T} Q_i (\hat{y}_{k+i} - r_{k+i})}_{\text{I}} + \underbrace{\sum_{i=1}^{n_c} (\hat{u}_{k+i-1}^T R_i \hat{u}_{k+i-1})}_{\text{II}} + \underbrace{\sum_{i=1}^{n_c} (\Delta \hat{u}_{k+i-1}^T R_{\Delta i} \Delta \hat{u}_{k+i-1})}_{\text{III}} \tag{4.14}$$

Or via in the Eq. 4.11 and 4.12 introduced algebraic vector notation:

$$J = \underbrace{(\bar{y}_{k+1} - \bar{r}_{k+1})^T Q (\bar{y}_{k+1} - \bar{r}_{k+1})}_{\text{I}} + \underbrace{\bar{u}_k^T R \bar{u}_k}_{\text{II}} + \underbrace{\Delta \bar{u}_k^T R_\Delta \Delta \bar{u}_k}_{\text{III}} \tag{4.15}$$

, with $Q \in R^{p \cdot n_p x p \cdot n_p}$, $R \in R^{m \cdot n_c x m \cdot n_c}$ and ${R_\Delta}^{m \cdot n_c x m \cdot n_c}$
Control error-, input-, and input rate signal are weighted with the matrices Q, R and R_Δ, respectively. Making an appropriate choice of these matrices is a significant segment of the MPC design. Input weighting terms R and R_Δ prevent the undesired, eruptive changes and magnitudes of the input signal. MPC tuning (choice of weighting terms) is initially performed in the simulation and than finalized on a real plant. For that matter, positive-definite, diagonal matrices are considered as reasonable choice.

Note: The objective function presented in Eq. 4.15 is not a unique choice. Beside quadratic, objective function can take a linear form as well. Other signals, e.g. system states x or output y, instead of control error e and input u, can be introduced as the penalizing terms. Furthermore, input signal vector u can be considered as the optimization variable instead of the input signal rate Δu. Eq. 4.13 can be rewritten to a following form:

$$\bar{y}_{k+1} - \bar{r}_{k+1} = \underbrace{\overbrace{Fx_k + Gu_{k-1} + G_d d_k}^{\bar{g}_k} - \bar{r}_{k+1}}_{\bar{e}_k} + H\Delta\bar{u}_k$$

$$\begin{aligned} J(\mathrm{I}) &= (\bar{e}_k + H\Delta\bar{u}_k)^T Q(\bar{e}_k + H\Delta\bar{u}_k) \\ &= \dots \\ &= 2\bar{e}_k^T QH\Delta\bar{u}_k + \Delta\bar{u}_k^T H^T QH\Delta\bar{u}_k \\ J(\mathrm{II}) + J(\mathrm{III}) &= (Lu_{k-1} + M\Delta\bar{u}_k)^T R(Lu_{k-1} + M\Delta\bar{u}_k) + \Delta\bar{u}_k^T R_\Delta \Delta\bar{u}_k \\ &= \dots \\ &= 2u_{k-1}^T L^T RM\Delta\bar{u}_k + \Delta\bar{u}_k^T M^T RM\Delta\bar{u}_k + \Delta\bar{u}_k^T R_\Delta \Delta\bar{u}_k \end{aligned}$$

$$J = \Delta\bar{u}_k^T (H^T QH + R_\Delta + M^T RM)\Delta\bar{u}_k + 2(\bar{e}_k^T QH + u_{k-1}^T L^T RM)^T \Delta\bar{u}_k \tag{4.16}$$

4.5.3 Consideration of Constraints

As already declared in the MPC introduction, a real strength of this controller is a simple and straightforward consideration of constraints. In Eq. 4.17 introduced input-, input rate-, state- and output constraints can be reduced to the input rate constraints.

$$u_{min} \le \hat{u}_{k+i} \le u_{max} \quad \forall \quad 0 \le i \le n_c - 1 \qquad \hat{u}_k \in R^m \tag{4.17a}$$
$$\Delta u_{min} \le \Delta\hat{u}_{k+i} \le \Delta u_{max} \quad \forall \quad 0 \le i \le n_c - 1 \qquad \Delta\hat{u}_k \in R^m \tag{4.17b}$$
$$x_{min} \le \hat{x}_{k+i} \le x_{max} \quad \forall \quad 1 \le i \le n_p \qquad \hat{x}_k \in R^n \tag{4.17c}$$
$$y_{min} \le \hat{y}_{k+i} \le y_{max} \quad \forall \quad 1 \le i \le n_p \qquad \hat{y}_k \in R^p \tag{4.17d}$$

In the Eq. 4.17 introduced relations can be rewritten in the algebraic vector notation and expressed as a function of the optimization variable $\Delta\bar{u}_k$.

1) Input Constraints (Eq. 4.17a)

$$\begin{aligned} \bar{u}_{min} \le &\bar{u}_k \le \bar{u}_{max} \\ \bar{u}_{min} \le Lu_{k-1} + M\Delta\bar{u}_k &\le \bar{u}_{max} \end{aligned}$$

Left side:

$$\begin{aligned} \bar{u}_{min} - Lu_{k-1} &\le M\Delta\bar{u}_k \\ -M\Delta\bar{u}_k &\le Lu_{k-1} - \bar{u}_{min} \end{aligned}$$

Right side:

$$M\Delta\bar{u}_k \le \bar{u}_{max} - Lu_{k-1}$$

Final formulation:

$$A_u \Delta \bar{u}_k \leq b_u$$

$$A_u = \begin{bmatrix} -M \\ M \end{bmatrix} \qquad b_u = \begin{bmatrix} -\bar{u}_{min} + Lu_{k-1} \\ \bar{u}_{max} - Lu_{k-1} \end{bmatrix}$$

2) Input Rate Constraints (Eq. 4.17b)

$$\Delta \bar{u}_{min} \leq \Delta \bar{u}_k \leq \Delta \bar{u}_{max}$$

Left side:

$$I \Delta \bar{u}_k \leq \Delta \bar{u}_{max}$$

Right side:

$$-I \Delta \bar{u}_k \leq -\Delta \bar{u}_{min}$$

Final formulation:

$$A_{\Delta u} \Delta \bar{u}_k \leq b_{\Delta u}$$

$$A_{\Delta u} = \begin{bmatrix} -I \\ I \end{bmatrix} \qquad b_{\Delta u} = \begin{bmatrix} -\Delta \bar{u}_{min} \\ \Delta \bar{u}_{max} \end{bmatrix}$$

3) Output Constraints (Eq. 4.17c)

$$\bar{y}_{min} \leq \bar{y}_{k+1} \leq \bar{y}_{max}$$
$$\bar{y}_{min} \leq Fx_k + Gu_{k-1} + G_d d_k + H \Delta \bar{u}_k \leq \bar{y}_{max}$$

Left side:

$$-H \Delta \bar{u}_k \leq -\bar{y}_{min} + Fx_k + Gu_{k-1} + G_d d_k$$

Right side:

$$H \Delta \bar{u}_k \leq \bar{y}_{max} - Fx_k - Gu_{k-1} - G_d d_k$$

Final formulation:

$$A_y \Delta \bar{u}_k \leq b_y$$

$$A_y = \begin{bmatrix} -H \\ H \end{bmatrix} \qquad b_y = \begin{bmatrix} -\bar{y}_{min} + Fx_k + Gu_{k-1} + G_d d_k \\ \bar{y}_{max} - Fx_k - Gu_{k-1} - G_d d_k \end{bmatrix}$$

4) State Constraints (Eq. 4.17d)

$$\bar{x}_{min} \leq \bar{x}_{k+1} \leq \bar{x}_{max}$$
$$\bar{x}_{min} \leq F_x x_k + G_x u_{k-1} + G_{d,x} d_k + H_x \Delta \bar{u}_k \leq \bar{y}_{max}$$

Left side:	Right side:
$-H_x\Delta\bar{u}_k \leq -\bar{x}_{min} + F_x x_k$	$H_x\Delta\bar{u}_k \leq \bar{x}_{max} - F_x x_k$
$+G_x u_{k-1} + G_{d,x} d_k$	$-G_x u_{k-1} - G_{d,x} d_k$

Final formulation:

$$A_x\Delta\bar{u}_k \leq b_x$$

$$A_x = \begin{bmatrix} -H_x \\ H_x \end{bmatrix} \qquad b_x = \begin{bmatrix} -\bar{x}_{min} + F_x x_k + G_x u_{k-1} + G_{d,x} d_k \\ \bar{x}_{max} - F_x x_k - G_x u_{k-1} - G_{d,x} d_k \end{bmatrix}$$

Finally, individually formulated constraints can be assembled to:

$$A\Delta\bar{u}_k \leq b$$

$$A = \begin{bmatrix} A_u \\ A_{\Delta u} \\ A_y \\ A_x \end{bmatrix} \quad \text{and} \quad b = \begin{bmatrix} b_u \\ b_{\Delta u} \\ b_y \\ b_x \end{bmatrix} \tag{4.18}$$

4.5.4 MPC Implementation

The objective function- and constraints formulations introduced in the Eq. 4.16 and 4.18 correspond to the quadratic programming problem.

Quadratic Programming Problem

$$\begin{aligned} \min_x \quad & \frac{1}{2} x^T Q x + c^T x \\ s.t. \quad & Ax \leq b \end{aligned} \tag{4.19}$$

Eq. 4.19 reads as: Find the optimal solution $x^\star$, minimizing the quadratic function $\frac{1}{2}x^T Q x + c^T x$, in the feasible region of x where the linear equality constraints $Ax \leq b$ are fulfilled. Algebraic vector $\Delta\bar{u}_k$ corresponds to the optimization variable x, and the matrices can be evaluated as follows:

$$Q = 2(H^T Q H + R_\Delta + M^T R M) \qquad \text{and} \qquad c^T = (\bar{e}_k^T Q H + u_{k-1}^T L^T R M)^T$$

Inequality matrices A and b correspond to the in the Eq. 4.18 introduced matrices. Optimization problem is solved by means of the numerical optimization solver qpOASES [Hans Joachim Ferreau, 2014]. Outcome of the optimization is the input signal rate sequence $\Delta\bar{u}_k$.

4.5.5 Luenberger Observer

One of the requirements for the MPC design is the current system state x_k. In that sense, when the system state has no physical meaning or cannot be measured, one needs to design and implement an observer.
Trivial observer represents a simple a copy of the system of interest. In that case, the dynamics of estimation error cannot be affected. Therefore, a Luenberger observer is implemented. Luenberger observer employs the identified state-space-model of the system, and adds a correction term weighting the output estimation error to it.

Plant model:	Trivial Observer:	Luenberger Observer:
$\dot{x} = Ax + bu$	$\dot{\hat{x}} = A\hat{x} + bu$	$\dot{\hat{x}} = A\hat{x} + bu + L(y - \hat{y})$
$y = c^T x$	$\hat{y} = c^T \hat{x}$	$\hat{y} = c^T \hat{x}$
	$\dot{e} = Ae$	$\dot{e} = (A - Lc^T)e$

Error dynamic can be influenced by the choice of the weighting term L. L is chosen so that the eigenvalues of estimation error system matrix $(A - Lc^T)$ lie in the left half plane.
Luenberger observer can be discretized in a straightforward manner. In the time-discrete domain, L is chosen so that the eigenvalues of error system matrix lie inside of the unit circle.

Plant model:	Luenberger Observer:
$x_{k+1} = Ax_k + bu_k$	$\hat{x}_{k+1} = A\hat{x}_k + bu_k + L(y_k - \hat{y}_k)$
$y_k = c^T x_k$	$\hat{y}_k = c^T \hat{x}_k$
	$e_{k+1} = (A - Lc^T)e_k$

Table 4.2: Overview of control parameters of TP MPC

n_p	n_c	Q_i	R_i	$R_{\Delta i}$
80	80	10000	$\begin{bmatrix} 0.3 & 0 \\ 0 & 0.01 \end{bmatrix}$	$\begin{bmatrix} 0.5 & 0 \\ 0 & 0.1 \end{bmatrix}$

4.6 TP Fill Level Controller

The second control objective is the production of tablets with constant properties. Empirical results have shown, that if the TP fill level sinks below a certain value, it has a significant impact on the quality of the produced tablets, i.e. it invokes a non neglectable quality degradation. On the other hand, variations of TP turret speed within a certain range are welcomed, as they do no affect the quality of final product.
TP MPC is utilized to ensure that the TP fill level stays possibly close to its reference value. A starting point for the MPC is in the Section 3.2.4 determined model of the DC line. This model joins the identified blender and TP model, and depends on the DC discharge signal. If discharge is activated, coupling between the models vanishes, and only the TP turret speed can be considered as the manipulated variable. Otherwise, TP MPC operates with two manipulated variables, blender inlet mass flow and TP turret speed. Although, the change of blender inlet mass flow could affect the blend uniformity, this effect can be neglected by the nominal API concentration.
MPC objective function is formulated as suggested introduced in the Eq. 4.16. Initial controller tuning, i.e. the choice of parameters Q, R, R_Δ, n_p and n_c, is performed in the simulation, and further improved on the real system. Another requirement for the MPC application is a current state vector x_k. As a state vector of DC model has no physical meaning, a Luenberger observer is employed for the state estimation. Fig. 4.12 provides a schematic representation of the TP fill level controller.

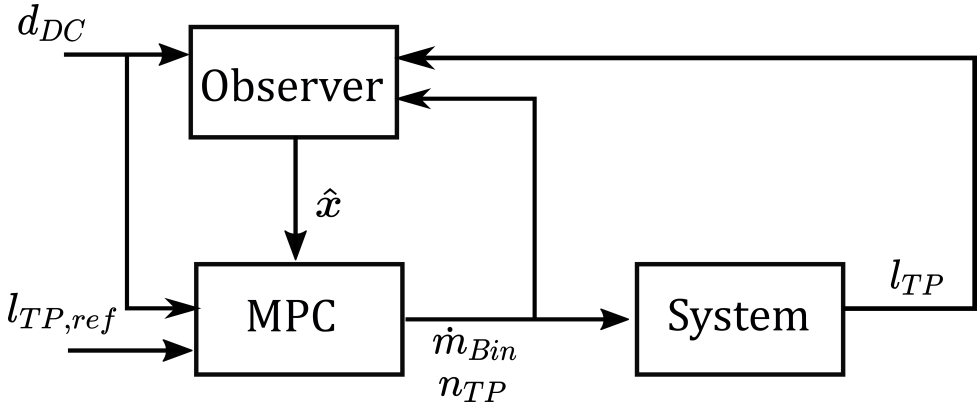


Figure 4.12: Schematic overview of TP fill level controller

4.7 DC-Line Controller

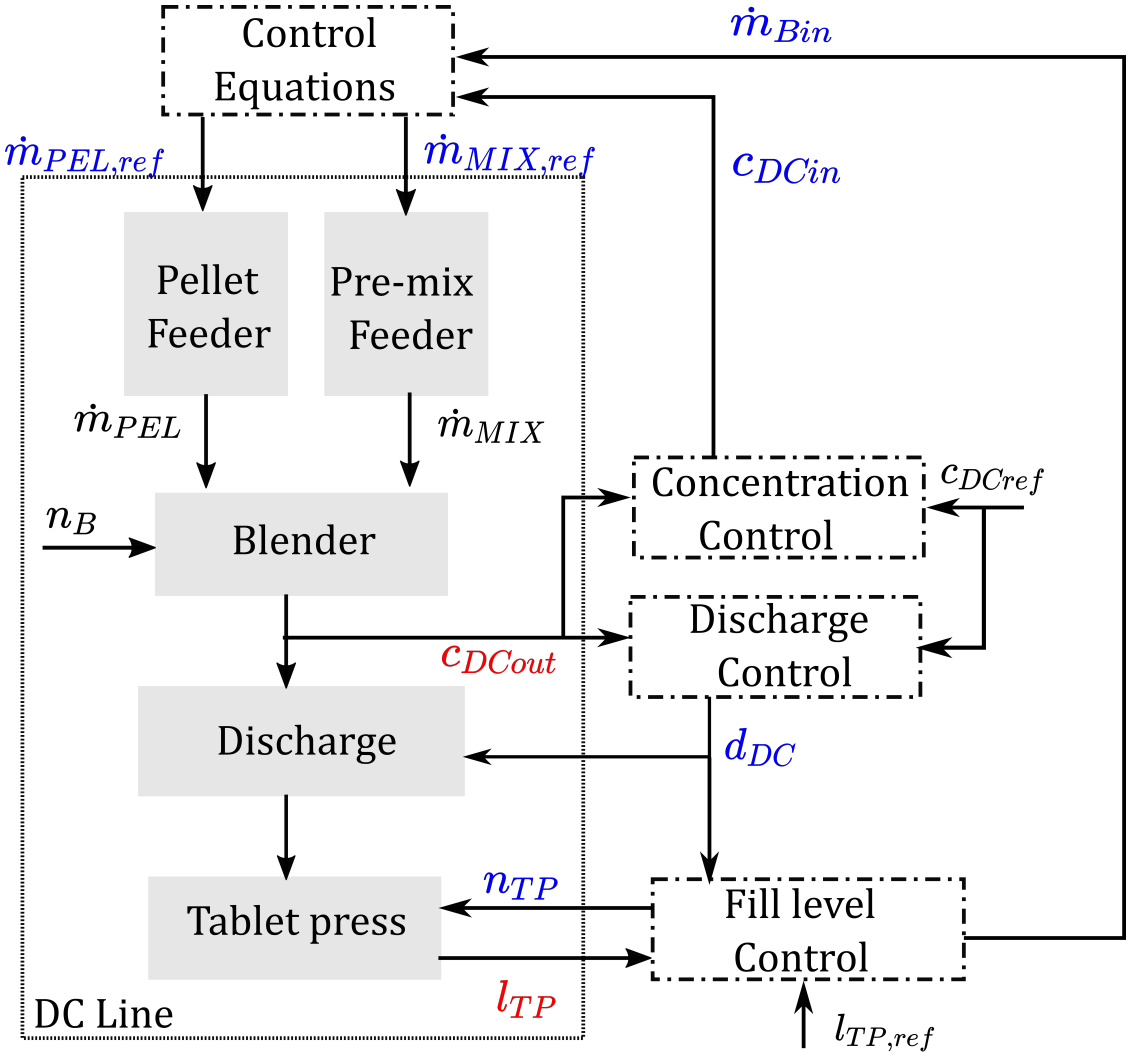


Figure 4.13: Schematic representation of DC line controller: Red and blue color depict measured/-controlled and manipulated system variables, respectively. Individual unit operations are indicated with gray background and the control blocks with dash-dot lines.

This Section outlines a development of a DC Line control concept. This concept combines the concentration-, discharge- and TP fill level controller. First control objective, i.e. keeping the API concentration in the allowed range, is addressed via application of the concentration controller. If the material exhibits unsuitable API properties, it is discharged by means of the discharge controller. TP MPC ensures, that the TP fill level stays close to the nominal point. Concentration controller adjusts the blender inlet concentration $c_{DCin,ref}$. At the same time, TP MPC manipulates the blender inlet mass flow. These two quantities are consigned to the *Control Equations* block, which reads as:

$$\dot{m}_{PEL,ref} + \dot{m}_{MIX,ref} = \dot{m}_{Bin,ref} \qquad \frac{\dot{m}_{PEL,ref}}{\dot{m}_{PEL,ref} + \dot{m}_{MIX,ref}} = c_{DCin,ref}$$

$$\longrightarrow$$

$$\dot{m}_{PEL,ref} = c_{DCin,ref} \cdot \dot{m}_{Bin,ref} \qquad \dot{m}_{MIX,ref} = \dot{m}_{Bin,ref} \cdot (1 - c_{DCin,ref})$$

Additionally, TP MPC manipulates the TP turret speed. Magnitude of MPC input signal is constrained with $u_{min} = [-1 \quad -20]$ and $u_{max} = [5 \quad 20]$, i.e. TP turret speed may be varied $\pm 20 rpm$ around the nominal point and blender inlet mass flow must stay in the range between $[4kg/h \quad 9kg/h]$. Additionally, TP turret speed rate is bounded by $\pm 1 rpm/s$. Fig. 4.13 provides a schematic overview of DC line controller.

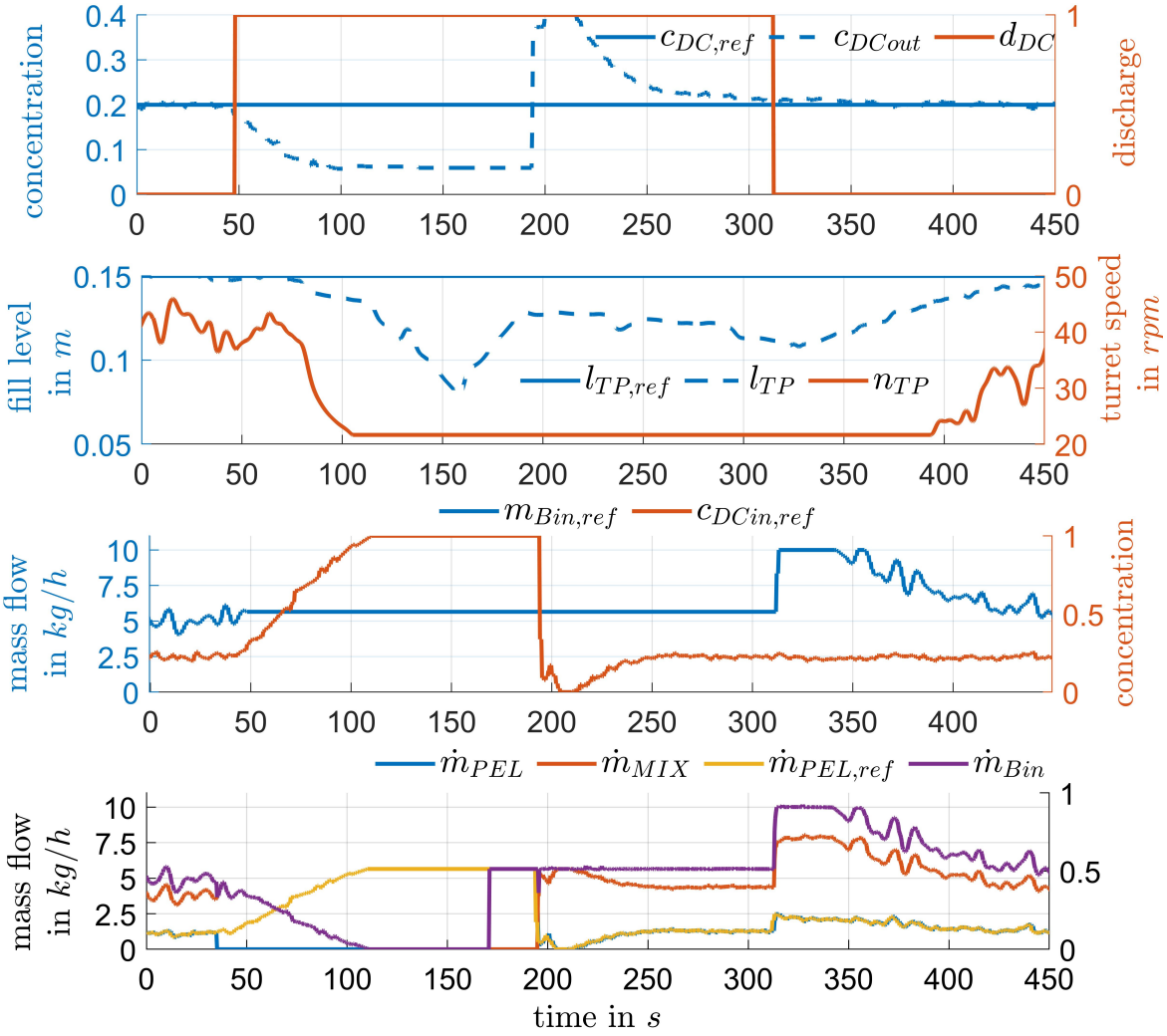


Figure 4.14: Investigation of DC line controller by pellet feeder failure on the real system.

The functionality of the DC line controller is examined via two test scenarios. First test scenario involves an artificially introduced 125 seconds long pellet feeder failure. This implies, that independent from the set point, pellet feeder mass flow equals zero. Consequently, concentration at the blender outlet rapidly decreases and discharge signal is activated. Concentration controller increases the pellet feeder set point, however this leads to no improvement, as the pellet feeder

mass flow equals zero. After approximately 100 seconds, the NIR measured concentration stays constant, which implies that the concentration sinks below the measurement range of the NIR device. At the same time, due to the discharge activation, no material enters the TP hopper. Consequently, TP fill level decreases. MPC copes with this deviation by decreasing the TP turret speed and increasing the blender inlet mass flow, as soon as the discharge is deactivated. Blender outlet concentration exhibits a large overshoot, due to the discrepancy between the real plant and Smith predictor input. As the API concentration decreases, concentration controller increases the blender inlet concentration. Smith predictor utilizes the inlet concentration set by the controller as the input signal. Yet, this value does not correspond to the real blender inlet concentration. This inconsistency can be improved by modification of the Smith predictor input to zero or one by the detection of pre-mix or pellet feeder failure. The introduced test scenario is examined both, in the simulation and on the real system. Both implementations lead to the similar results. The obtained results once more verify the quality of the identified models and confirm the correct implementation of control algorithms on the real system.

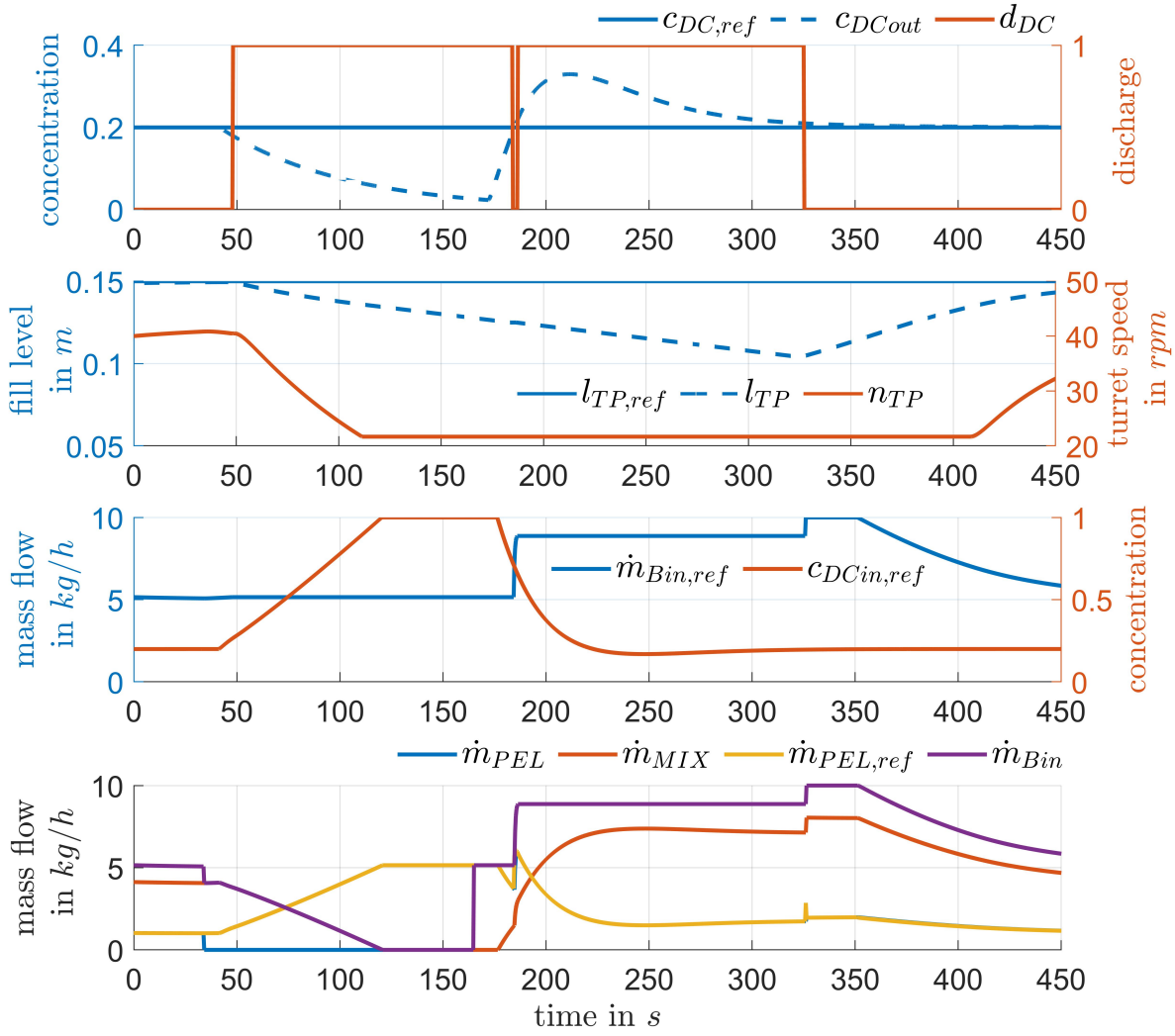


Figure 4.15: Investigation of DC line controller by pellet feeder failure in the simulation.

Second test scenario involves an intentional API contamination. At approximately 40. second of the test scenario, pellet feeder is contaminated with the pre-mix material. Consequently, the concentration at the blender outlet decreases and discharge signal is activated. Concentration controller tends to keep the concentration within the allowed boundaries by increasing the mass flow of the pellet feeder. Due to the discharge activation, no material enters the TP, and the TP fill level decreases. TP MPC decreases the turret speed and increases the blender inlet mass flow, as soon as discharge is deactivated. It takes approximately 180 seconds until the plant is back to the nominal operating points. In this test scenario, input signal rate constraint is discarded.

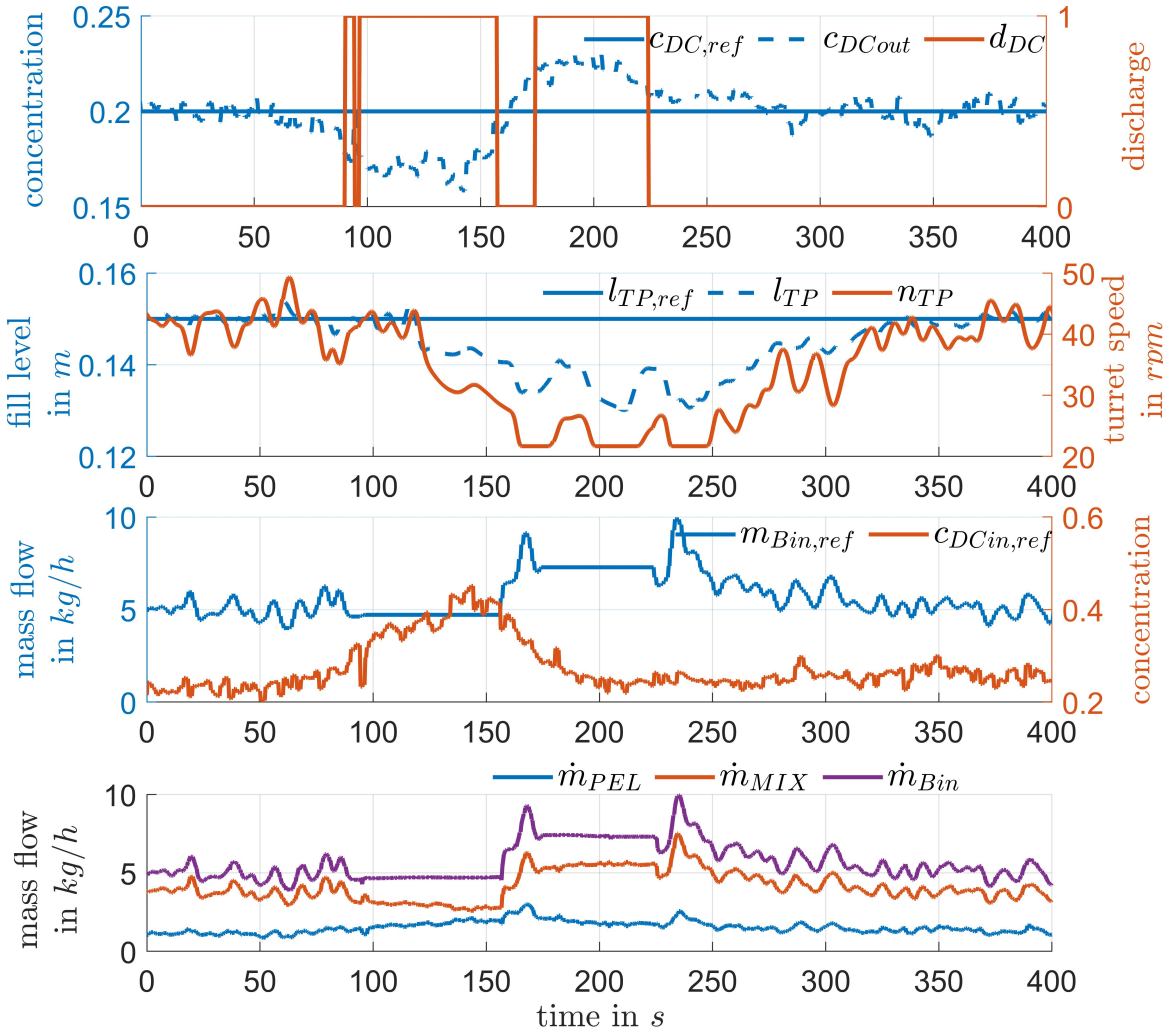


Figure 4.16: Investigation of DC line controller by API contamination on the real system.

5 Control Concept for Complete Production Plant

This Chapter outlines a development of a control law for the complete continuous tablet manufacturing plant. Three different candidates for the master unit of the complete plant are suggested: Extruder (as a master unit of Concept A), tablet press (as a master unit of Concept B) and no master unit (Concept C). Depending on the active master unit, choice of manipulated variables of individual controllers change, while the control objectives stay the same.
First control objective is to keep the API concentration within the allowed range around the reference during the production. As there was only one NIR device for the concentration measurement available at the time, it was decided to utilize it within the HME line. In that sense, it is not possible to implement the feedback control of blender outlet concentration in the DC line. Mass flow rates of pellet and pre-mix feeder are adjusted so, that the resulting blender inlet concentration corresponds to the reference value. Discharge signal is triggered using the blender outlet concentration predicted via identified DC concentration transfer function. Concentration controller in the HME line manipulates the extruder inlet concentration by adjusting the mass flow rates of API and polymer feeder via control equations. Discharge signal is triggered by the NIR concentration measurement. However, when two NIR sensors for the concentration measurement would be available, both developed concentration controllers could be implemented simultaneously in the continuous plant operation. Concentration control is realized in the same manner independently from the active master unit.
TP MPC copes with the second control objective, i.e. keeping the TP fill level close to the reference value. Depending on the active master unit, MPC manipulates TP turret speed and/or blender inlet mass flow.
Within the HME line produced pellets are gathered in the pellet hopper and proceeded to the pellet feeder via pneumatic transport unit. This leads to the third control objective, i.e. prevention of the undesired process events, such as feeder or hopper exhaustion/overflow. In that sense, total mass flow rates of HME or DC line are manipulated via mass flow controller. Additionally, mass flow controller manipulates the transport activation signal based on the current hold-up in the pellet feeder. If the pellet feeder runs below a certain threshold, transport is activated. However, even if the pellet hold-up permanently stays below the specified threshold, transport cannot be activated more than once every two minutes, or if the HME discharge signal is activated. Transport activation condition stays the same in each designed control concepts.
Control concepts A, B and C differ on the choice of manipulated variables for TP MPC and mass flow controller. Detailed explanation of each of these concepts is provided in the Sections 5.1, 5.2 and 5.3.
In order to examine the developed control concepts, a complete continuous plant

is built up in the simulation. Formerly identified models of feeding, extrusion, cooling&pelletisation, blending and TP, are utilized for that purpose. Additionally, simple mass balance relations for the simulation of pellet hopper and pellet feeder are developed in the Section 5.4. Finally, control concepts A, B and C are designed and examined via same test scenario in the simulation. Simulation results are presented in the Fig. 5.3, 5.4 and 5.5, and the designed control concepts are compared regarding the essential process properties.

5.1 Control Concept A: Extruder

Control concept A, with the extruder as a master unit, tends to keep the total mass flow in the HME line constant. This is ensured by setting the extrusion inlet mass flow to its nominal value $\dot{m}_{Ein} = 2kg/h$. At the same time, concentration controller manipulates the extrusion inlet concentration c_{HMEin}. These two quantities are consigned to the *Control Equations 1* block, leading to the explicit set points for the pellet and polymer feeder mass flow rates. The *Control Equations 1* block reads as:

$$\dot{m}_{API,ref} + \dot{m}_{P0L,ref} = \dot{m}_{Ein} \qquad \frac{\dot{m}_{API,ref}}{\dot{m}_{API,ref} + \dot{m}_{P0L,ref}} = c_{HMEin}$$

$$\longrightarrow$$

$$\dot{m}_{API,ref} = c_{HMEin} \cdot \dot{m}_{Ein} \qquad \dot{m}_{P0L,ref} = \dot{m}_{Ein} \cdot (1 - c_{HMEin})$$

Mass flow controller is utilized to attain the uniform material transport between the two lines. It tends to keep the constant time intervals between the pellet transport activations by manipulating the pellet feeder mass flow. Transportation time interval needs to be chosen carefully, as each discharge activation in the HME line leads to the emptying of the pellet hopper. In that sense, an excessively long time interval could lead to the losses of valuable material. Therefore, transportation time interval is set to $t_T = 240s$.
Change of the pellet feeder internal hold-up Δm_{PEL} is measured by each pellet transport activation, and mass flow set point is calculated as:
$\dot{m}_{PEL,ref} = \frac{3600}{t_T} \Delta m_{PEL}$
Pellet feeder set point stays constant between the pellet transport activations. Pre-mix feeder mass flow is adjusted via *Control Equations 2* block so that the resulting blender inlet concentration corresponds to the reference value. The *Control Equations 2* block reads as:

$$\frac{\dot{m}_{PEL,ref}}{\dot{m}_{PEL,ref} + \dot{m}_{MIX,ref}} = c_{DCin}$$

$$\dot{m}_{MIX,ref} = \dot{m}_{PEL,ref} \cdot \left(\frac{1}{c_{DCin}} - 1\right)$$

In that sense, blender inlet mass flow, which corresponds to the sum of single feeder flow rates, is already fix, and TP MPC can manipulate solely the TP turret speed. In this case, TP inlet mass flow is considered as the known disturbance, which is predicted via blender model identified in the Chapter 2, Subsection 3.2.2. A schematic representation of the concept A control structure is depicted in the Fig. 5.1.

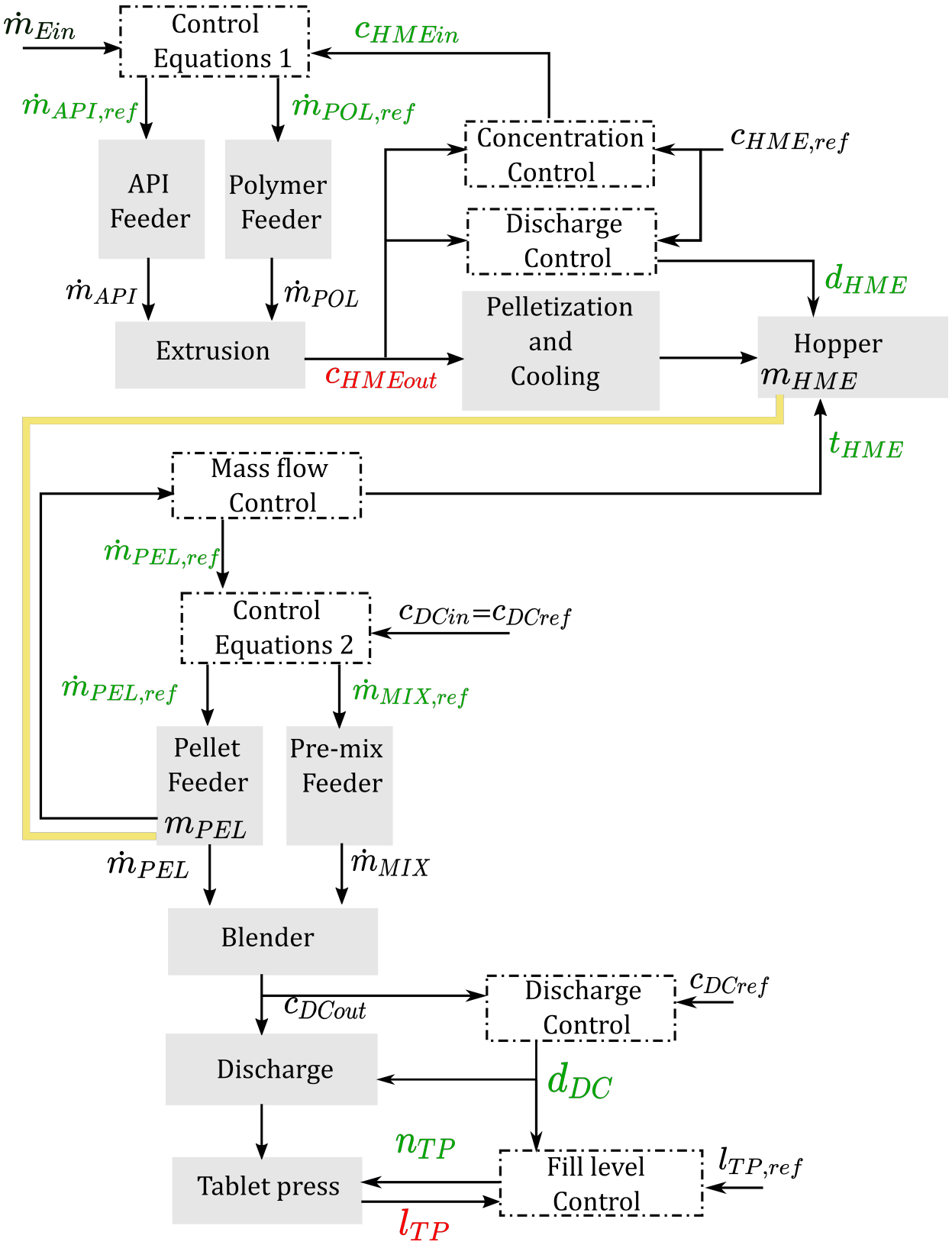


Figure 5.1: Schematic representation of control concept A: Red and green color depict measured/-controlled and manipulated system variables, respectively. Yellow line indicates the pellets transported from the pellet hopper to the pellet feeder. Individual unit operations are indicated with gray background and the control blocks with dash-dot lines.

5.2 Control Concept B: Tablet Press

Control concept B, with the TP as a master unit, tends to keep the TP turret speed constant. This is achieved via definition of an additional equality constraint in the MPC optimization problem. Again, TP MPC has only one degree of freedom, in the present case the blender inlet mass flow $\dot{m}_{Bin}$. Blender inlet concentration is set to the reference value. Again, these two quantities are consigned to the *Control Equations 2* block, which now reads as:

$$\dot{m}_{PEL,ref} + \dot{m}_{MIX,ref} = \dot{m}_{Bin} \qquad \frac{\dot{m}_{PEL,ref}}{\dot{m}_{PEL,ref} + \dot{m}_{MIX,ref}} = c_{DCin}$$

$$\longrightarrow$$

$$\dot{m}_{PEL,ref} = c_{DCin}\dot{m}_{Bin} \qquad \dot{m}_{MIX,ref} = \dot{m}_{Bin}(1 - c_{DCin})$$

Blender inlet mass flow, which corresponds to the total mass flow in the DC line, is manipulated via TP MPC. Therefore, in order to guarantee reliable feeding, i.e. no exceeding or running out of the material in the pellet feeder, total mass flow in the HME line needs to be adjusted by means of the mass flow controller. Time interval between the consecutive pellet transport activations t_T, as well as during this time interval in the HME line produced and from the DC line consumed amount of the material, Δm_{HME} and Δm_{DC}, can be measured. Total mass flow in the HME line, i.e. extrusion inlet mass flow, is adjusted to compensate the difference between the produced and consumed material:

$$m_{comp} = \Delta m_{DC} + \frac{1}{2}(\Delta m_{DC} - \Delta m_{HME})$$
$$\dot{m}_{Ein} = \frac{3600}{t_T} m_{comp}$$

The HME line total mass flow set point stays constant between the single transport activations. Mass flow rates of API and polymer feeder are evaluated via *Control Equations 1*, which employs the extruder inlet mass flow set by the mass flow controller and extrusion inlet concentration set by the concentration controller.

5.3 Control Concept C: No master unit

Control concept C operates with no master unit and provides an additional degree of freedom for the TP MPC. In this control concept, TP MPC manipulates both, the blender inlet mass flow and the TP turret speed. Mass flow- and concentration controller are implemented in the exactly same manner as in the control concept B. In that sense, the only difference between the control concepts B and C is an additional manipulated variable for the TP MPC. A schematic representation of the control concepts B and C is depicted in the Fig. 5.2.

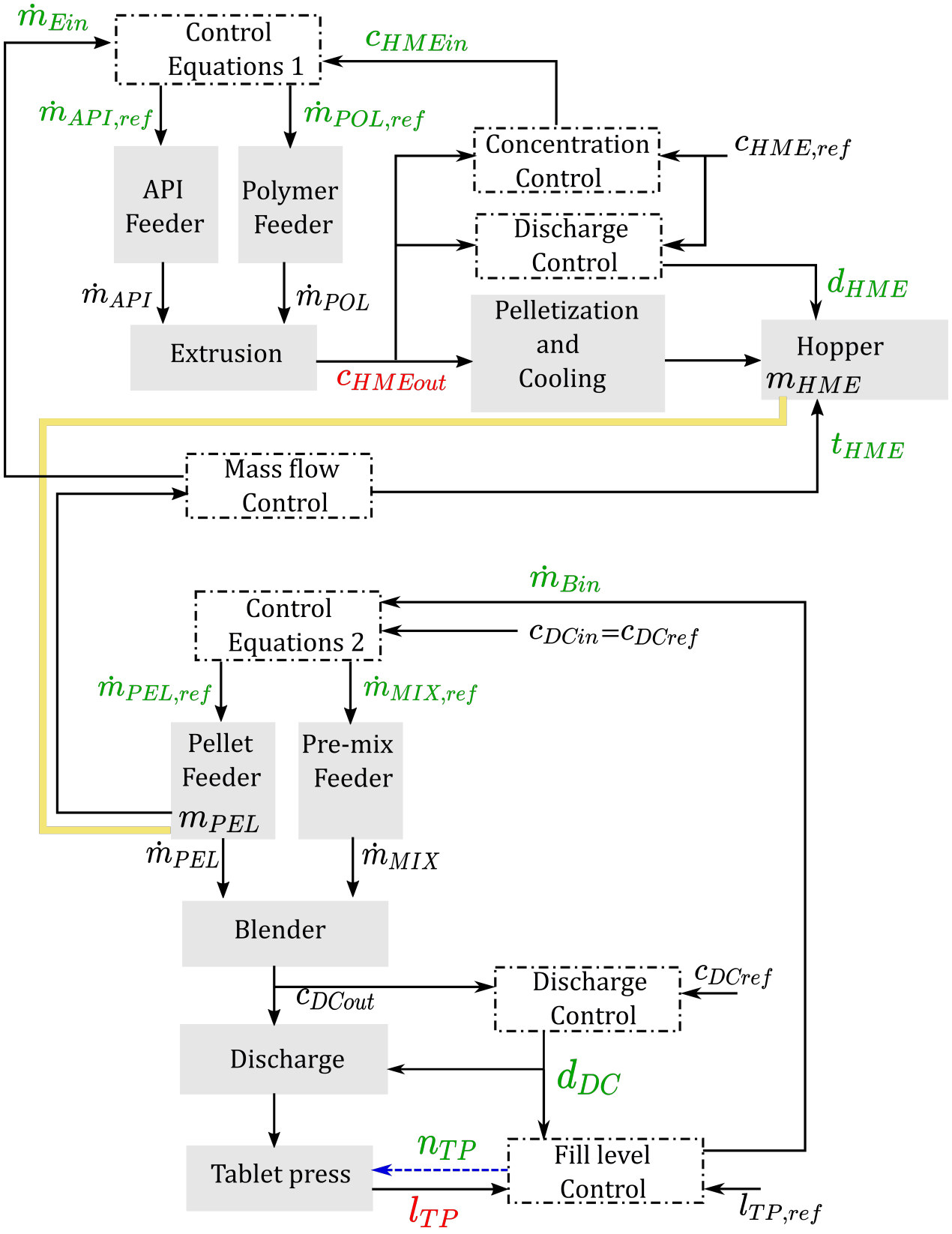


Figure 5.2: Schematic representation of control concept B & C: Red and green color depict measured/controlled and manipulated signals, respectively. Yellow line indicates the pellets transported from the pellet hopper to the pellet feeder. Single unit operations are indicated with gray background and individual control blocks with dash-dot lines. Blue dash line is valid only in the control concept B and indicates the only difference between the control concepts B and C.

5.4 Examination of Control Concepts in the Simulation

In order to examine the control action of the introduced control concepts in the simulation, the individual unit operations are substituted by the according modeling units. In addition to the in the Chapter 3 identified modeling units, pellet hopper- and pellet feeder hold-up need to be modeled for the simulation purposes. Pellet hopper hold-up m_{HME} can be described via simple mass balance equations. This computation employs the $9s$ delayed extrusion outlet mass flow $\dot{m}_{HME}$, i.e. the resulting mass flow after the pelletisation unit. If HME discharge- or pellet transport signal is activated, pellet hopper is emptied.

$$m_{HME,k=0} = 0kg$$
$$m_{HME,k} = (m_{HME,k-1} + \frac{T_s}{3600}\dot{m}_{HME,k}) \cdot (1 - d_{HME,k}) \cdot (1 - t_{HME,k})$$

Pneumatic conveying is triggered via pellet transport activation signal. Pellet feeder hold-up m_{PEL} computes as:

$$m_{PEL,k=0} = 1.3kg$$
$$m_{PEL,k} = m_{PEL,k-1} - \frac{T_s}{3600}\dot{m}_{PEL,k} + t_{HME,k}m_{HME,k-1}$$

Finally, a complete production plant can be built in the simulation. Functionality of the proposed control concepts is examined via same test scenario involving diverse feeder disturbances. In the first part, two API feeder failures are introduced. At the 20. minute of the simulation, real mass flow of the API feeder is set to be $+0.5kg/h$ more than the desired set point. At the 40. min, a complete API feeder failure occurs, i.e. the real mass flow is set to $0kg/h$. Both introduced disturbances last 1.5 minutes. Second part of the experiment involves the equivalent disturbances of the pellet feeder, occurring in the 75. and 95. minute of the simulation. The introduced disturbances lead to the deviations of API concentration, material discharge, and TP fill level deviations. In such a way, a control action of concentration, discharge, fill level and mass flow control concepts can be examined simultaneously. Fig. 5.3, 5.4 and 5.5 illustrate the process quantities of interest during the introduced test scenario.

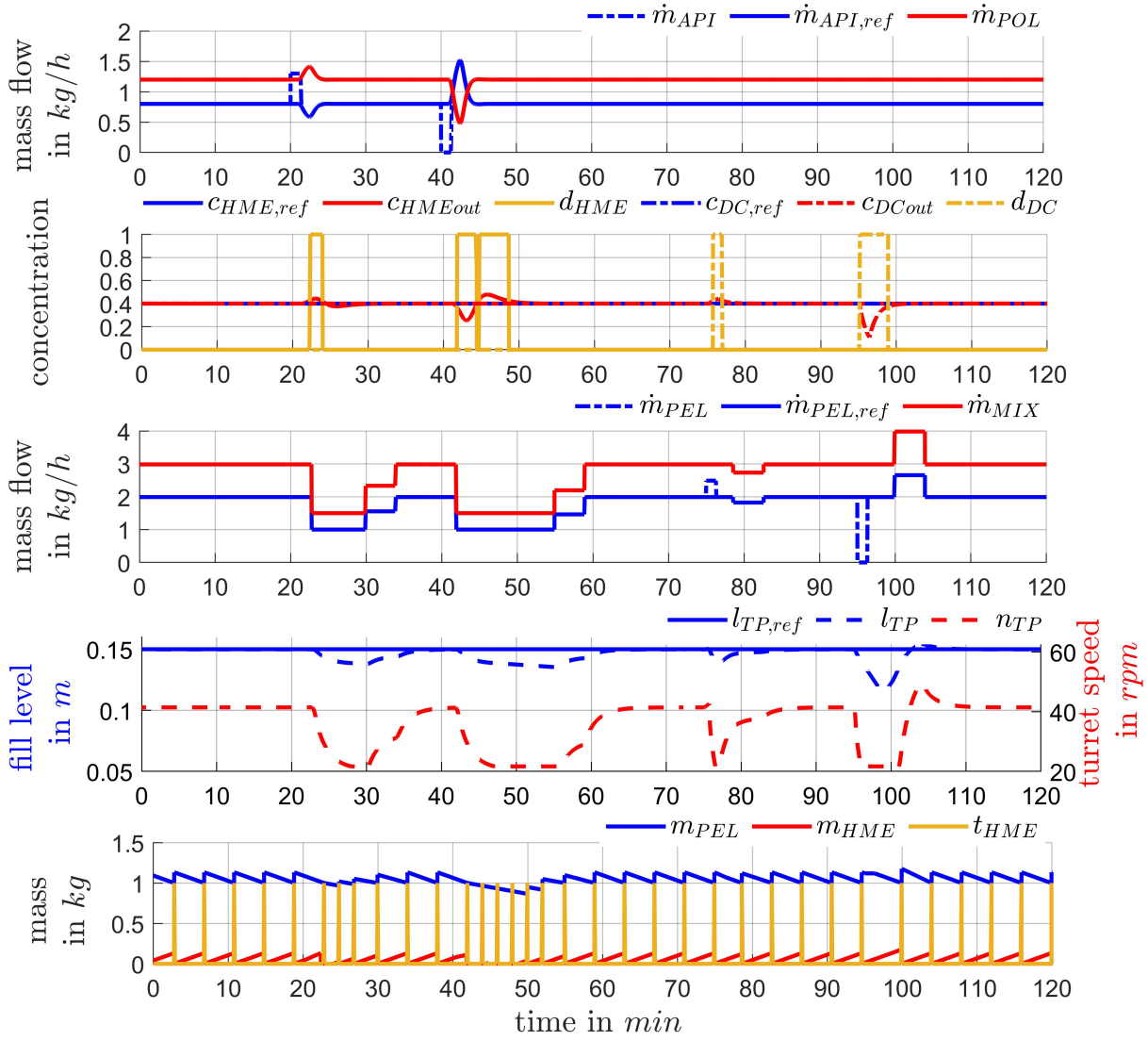


Figure 5.3: Investigation of control concept A in the simulation

Fig. 5.3 depicts the signals of interest during the introduced test scenario. API feeder disturbance, which manifests at the 20. simulation minute, results in the deviation of the extrusion outlet concentration. HME concentration controller copes with the disturbance by adjusting the ratio between the API and polymer feeder. Yet, as it does not accomplish to keep the concentration within the allowed range, HME discharge signal is activated and pellet hopper is emptied. After the deactivation of HME discharge, transport signal is activated. As there is no material in the pellet-hopper, no material enters the pellet feeder and hold-up decreases below $1kg$. In order to omit the material exhaustion, mass flow controller decreases the pellet feeder mass flow to the minimal value. As the mass flow ratio between pellet and pre-mix feeder stays constant, this leads to the decreased blender inlet mass flow, which then results in the decreased TP fill level. TP MPC copes with this deviation by decreasing the TP turret speed. After approximately 17 minutes, plant is back to the nominal operating points. A total API feeder failure occurring at the 40. simulation minute causes similar, only longer lasting deviations from the nominal operating points. It takes approximately 23 minutes, for the control concept to bring the plant back to the nominal operation. At the 75. minute of

simulation, a pellet feeder failure occurs, causing the deviation of concentration and discharge in the DC line. As no material enters the TP hopper, TP fill level decreases, and MPC reacts by adjusting the TP turret-speed. Introduced mass flow deviation leads to the increased consumption of the material in the pellet feeder. Due to that, mass flow controller decreases the pellet feeder mass flow by the next transport activation. It takes approximately 10 minutes until the plant is back to the nominal operating points. Finally, at the 95. minute of simulation, a complete failure of pellet feeder occurs. As there is no concentration controller active in DC line, concentration sinks almost to the half of its reference value. Discharge is activated, TP fill level decreases, and TP MPC adjusts the TP turret speed. This time, mass flow controller increases the pellet feeder mass flow in order to prevent the material overflow. Extrusion inlet mass flow, i.e. total mass flow in the HME line, stays constant during the test scenario and pellet transport takes place uniformly every four minutes in the nominal plant operation.

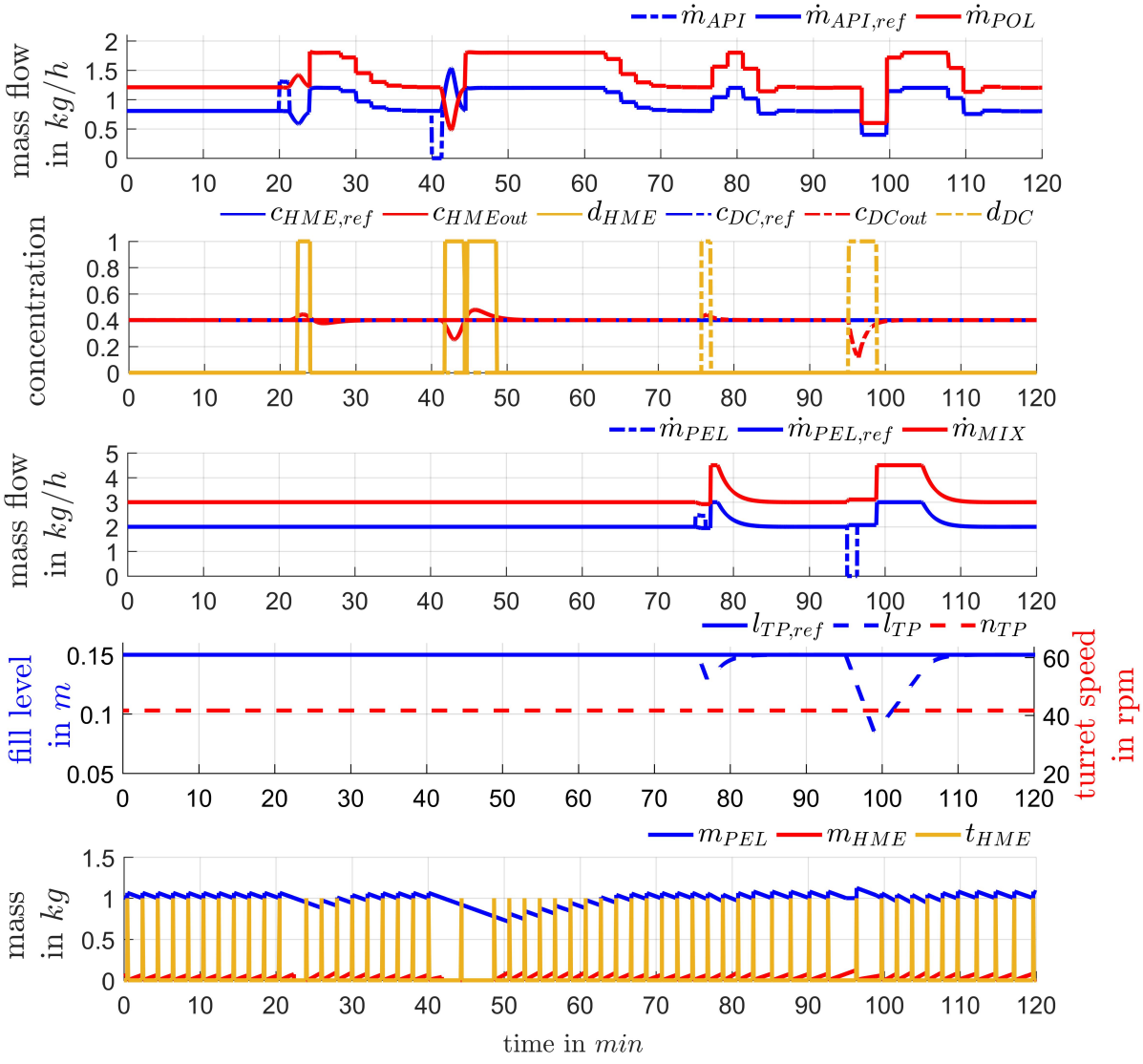


Figure 5.4: Investigation of control concept B in the simulation

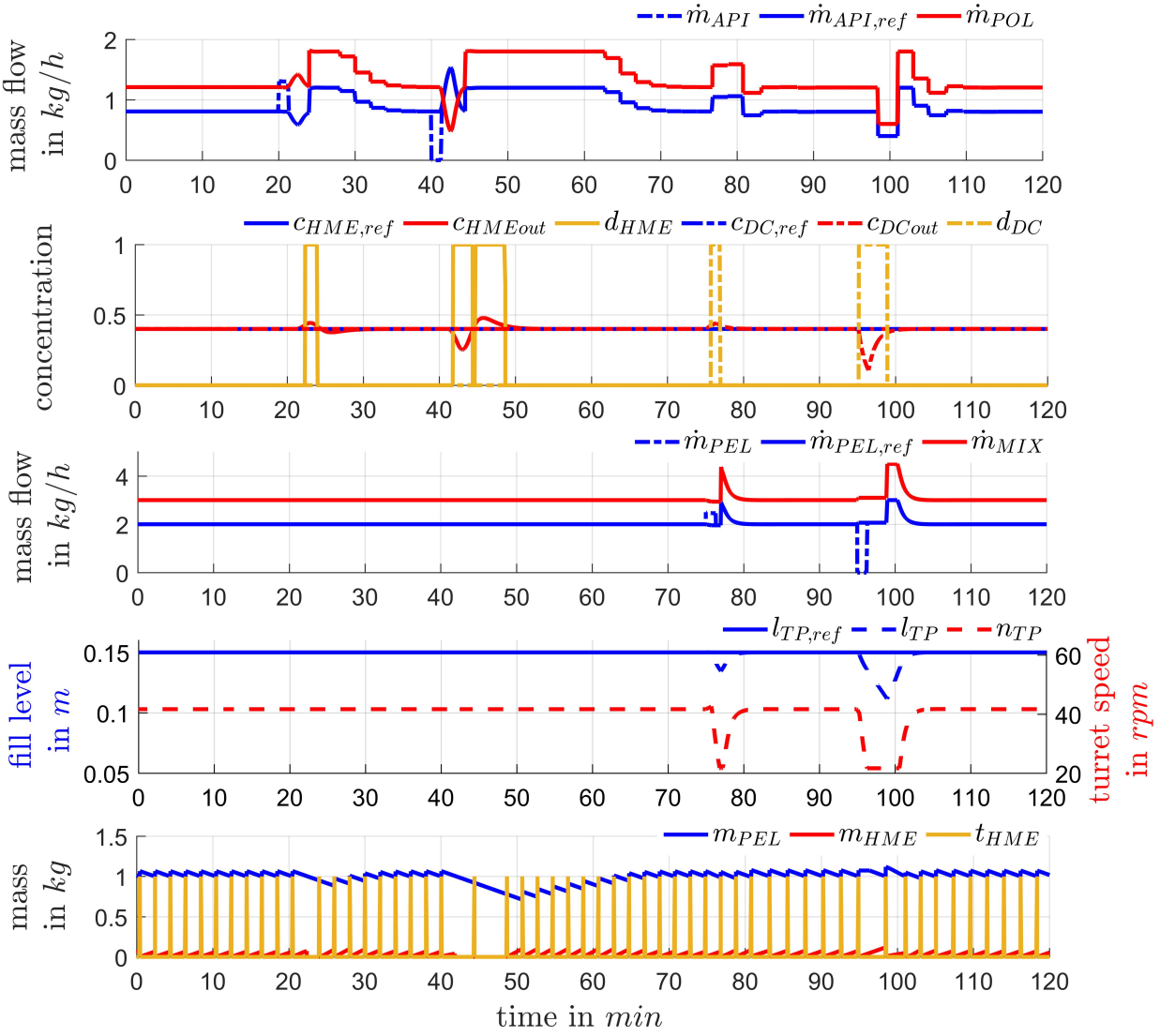


Figure 5.5: Investigation of control concept C in the simulation

Fig. 5.4 depicts the signals of interest during the introduced test scenario. API feeder disturbance occurs at the 20. minute of the simulation and results in the deviation of the HME line concentration. HME concentration controller adjusts the ratio between API and polymer feeder, yet does not accomplish to keep the concentration within the allowed range. Therefore, HME discharge signal is activated and pellet hopper is emptied. After the discharge deactivation, transport signal is activated, but as the pellet hopper is empty, no material enters the pellet feeder. Consequently, pellet feeder hold-up decreases below $1kg$. Mass flow controller increases the total mass flow in the HME line in order to prevent the material exhaustion. In contrast to the control concept A, blender inlet mass flow stays constant during the API disturbance, and no deviation of TP fill level occurs. After approximately 15 minutes, plant is back to the nominal operating points. At the 40. minute of simulation a total feeder failure occurs, again invoking the similar deviations. Total mass flow in the HME line is set to the maximal value and kept there, until the pellet hold-up returns above $1kg$. It takes approximately 25 minutes until the plant is brought back to the nominal operation. At the 75.

minute of simulation, a pellet feeder disturbance occurs, causes the deviation of concentration, and leads to the discharge in the DC line. As no material enters the TP hopper, TP fill-level decreases, and MPC reacts by adjusting the blender inlet mass flow, as soon as the discharge signal is deactivated. It takes approximately 5 minutes until the plant operates with the nominal values again. Finally, at the 95. minute of simulation, a complete failure of pellet feeder occurs. As there is no concentration controller active in DC line, concentration sinks far from the reference value. Discharge is activated, TP fill-level decreases, and TP MPC increases the blender inlet mass flow. In either case, an increased material consumption in the pellet feeder is compensated via mass flow controller increasing the total mass flow of the HME line. TP turret speed stays constant during the simulation.

Fig. 5.5 represents the signals of interest during the introduced test scenario. Process flow is very similar to the control concept B. Only difference between these two control concepts can be recognized is an additional degree of freedom for the TP MPC, i.e. TP turret speed. This leads to an improvement of the control action, and shorter deviations of TP fill level from the nominal value. In such a way, blender inlet mass flow exhibits less changes, leading to the more stable operation of the mass flow controller and the more uniform pellet transport between the lines.

The obtained simulation results are compared with regard to the critical process properties, i.e. total amount of the discharged material, and the duration of the fill level, concentration and mass flow deviations from the nominal operating points. As the implementation of the concentration controller is independent from the individual control concepts, each concept results in the same total deviation duration of 1.37 minutes. This duration could be furtherly reduced by the application of the concentration controller in the DC line.
Control concepts B and C exhibit the same amount of discharged material, i.e. $0.73kg$. Control concept A shows slightly better results with $0.65kg$ of discharged material.
When it comes to the TP fill level, control concept C shows superiority over the other control concepts with only 0.16 minutes long deviations from the nominal values. This result is expected, as only in this concept TP MPC manipulates both blender inlet mass flow and TP turret speed. Concepts A and B result in the similar duration of 0.51 and 0.46 minutes, respectively.
Regarding the mass flow deviations, control concept exhibits the least advantageous behavior with 80.42 minutes long deviations. Control concepts A and C are relative similar to one another, with total duration of 46.12 and 55.6 minutes, respectively.
In addition to these, a time until the continuous plant must stop by the complete pellet feeder failure is determined for each control concept. A TP fill level below $50mm$ is chosen as a condition for stopping the plant. Again, control concepts A and C result in almost same time of 10.9 and 10.7 minutes, respectively. Control concept B is inferior with only 6 minutes, until the production plant must stop. In that sense, the control concepts A and C would definitely be the candidates for the application on the real system.

6 Conclusion and Outlook

This master thesis provides a detailed description of the design, implementation and application of diverse control concepts for the continuous tablet manufacturing plant. Controller application results in the improved plant performance, keeping the API concentration in the wise range continuously along the plant, discharging the material of poor quality and producing the tablets with valid technical properties. These objectives are achieved by combining the standard and advanced control techniques, such as Smith predictor structure with PID and anti-windup for the concentration control, and MPC for the TP fill level control.
Individual plant unit operations, i.e. extrusion, cooling&pelletisation, blending and tablet compaction, are modeled separately. In that sense, the required laboratory experiments are executed on the real system and the collected data is utilized for the model identification purposes. Comparison of the estimated model and the real system leads to a conclusion, that the performed data-driven modeling provides satisfactory results and that the identified models can be employed for the controller development.
First developed control concept is a DC line controller, involving a TP MPC fill level control, discharge control via hysteresis controller, and a concentration control combining the Smith predictor structure and PID controller. This control concept is examined via different test scenarios, not only in the simulation, but on the real system, as well. Obtained results show a satisfactory compliance between the simulated and measured signals, confirming the accuracy of the identified models and successful application of control laws. Additionally, the obtained results provide the information regarding the potential improvement of the concentration controller, i.e. adjustment of Smith predictor input signal in case of the complete feeder failure. In summary, DC line control concept shows a very good disturbance rejection: TP fill level is successfully kept close to the reference by adjusting TP turret speed and blender inlet mass flow. Furthermore, API concentration stays in the wise range around the reference via manipulation of the blender inlet concentration. Tablet probes were taken during the artificially introduced test scenarios, i.e. feeder fault and API contamination. Results of the performed tablets analysis show very satisfactory results regarding the tablet properties. Potential next steps in the DC line would involve the utilization of other control techniques for the concentration control, such as algebraic synthesis or MPC.
In the final stage, a control concept for the complete continuous plant is developed. Additionally to the in the DC line introduced controllers, this control concept involves a concentration and discharge controller in the HME line, and a mass flow controller ensuring the reliable transport of material between the HME and the DC line. This control concept can be carried on with different master units, i.e. extruder (Control concept A), TP (Control concept B) or no master unit (Control concept C). Depending on the activated master unit, manipulated variables of single controllers change, while the control objectives stay the same. Control action of A, B and C is examined via same test scenario in the simulation. In summary, all

concepts show satisfactory behavior during the introduced API and pellet feeder disturbances. Yet, when considering the deviations from nominal plant operation, concept B shows less advantageous behavior compared to the A and C. Therefore, the control concepts A and C definitely are the reasonable candidates for the application on the real system. Anyway, due to the unreliable measurement of the concentration, the application on the real system is omitted. As soon as this issue is solved, developed control concepts can be implemented on the real system in the straightforward manner and examined via same test scenario as in the simulation. Thereby, the tablet probes should be taken and analyzed. Analysis results would show if the developed control concepts are suitable for further industrial application.

Bibliography

[Bhaskar et al., 2017] Bhaskar, A., Nunes de Barros, F., and Singh, R. (2017). Development and implementation of an advanced model predictive control system into continuous pharmaceutical tablet compaction process. *International Journal of Pharmaceutics*, 534.

[Celikovic et al., 2019] Celikovic, S., Kirchengast, M., Rehrl, J., Kruisz, J., Sacher, S., Khinast, J., and Horn, M. (2019). Model predictive control for continuous pharmaceutical feeding blending units. *Chemical Engineering Research and Design*, 154.

[evon, 2020] evon (2020). Evon gmbh. https://evon-automation.com/xamcontrol/. [Online; accessed 26-July-2020].

[Hans Joachim Ferreau, 2014] Hans Joachim Ferreau, Christian Kirches, A. P. H. G. B. M. D. (2014). qpoases: a parametric active-set algorithm for quadratic programming. *Math. Prog. Comp.*, 6:327–363.

[Horn and Dourdoumas, 2004] Horn, M. and Dourdoumas, N. (2004). *Regelungstechnik - Rechnerunterstützter Entwurf zeitkontinuierlicher und zeitdiskreter Regelkreise*. Pearson.

[Kruisz et al., 2017] Kruisz, J., Rehrl, J., Sacher, S., Aigner, I., Horn, M., and Khinast, J. (2017). Rtd modeling of a continuous dry granulation process for process control and materials diversion. *International Journal of Pharmaceutics*, 528(1-2):334–344.

[Lee et al., 2015] Lee, S., O'Connor, T., Yang, X., Cruz, C., Chatterjee, S., Madurawe, R., Moore, C., Yu, L., and Woodcock, J. (2015). Modernizing pharmaceutical manufacturing: from batch to continuous production. *Journal of Pharmaceutical Innovation*, 10.

[Nelles et al., 2000] Nelles, O., Fink, A., and Isermann, R. (2000). Local linear model trees (lolimot) toolbox for nonlinear system identification. *IFAC Proceedings Volumes*, 33(15):845 – 850. 12th IFAC Symposium on System Identification (SYSID 2000), Santa Barbara, CA, USA, 21-23 June 2000.

[Rehrl et al., 2018] Rehrl, J., Karttunen, A.-P., Nicolaï, N., Hörmann, T., Horn, M., Korhonen, O., Nopens, I., De Beer, T., and Khinast, J. (2018). Control of three different continuous pharmaceutical manufacturing processes: Use of soft sensors. *International Journal of Pharmaceutics*, 543.

[Rehrl et al., 2019] Rehrl, J., Kirchengast, M., Celikovic, S., Sacher, S., Kruisz, J., Khinast, J., and Horn, M. (2019). Improving pellet quality in a pharmaceutical hot melt extrusion process via pid control and lolimot-based mpc. *Journal of pharmaceutical innovation*.

[Rehrl et al., 2016] Rehrl, J., Kruisz, J., Sacher, S., Khinast, J., and Horn, M. (2016). Optimized continuous pharmaceutical manufacturing via model-predictive control. *International journal of pharmaceutics*, 510.

[Sacher et al., 2019] Sacher, S., Celikovic, S., Rehrl, J., Poms, J., Kirchengast, M., Kruisz, J., Sipek, M., Salar-Behzadi, S., Berger, H., Stark, G., Horn, M., and Khinast, J. (2019). Towards a novel continuous hme-tableting line: Process development and control concept. *European Journal of Pharmaceutical Sciences*, 142:105097.

[ssest, 2020] ssest (2020). MATLAB. https://de.mathworks.com/help/ident/ref/ssest.html. [Online; accessed 26-July-2020].

[Su et al., 2018] Su, Q., Moreno, M., Ganesh, S., Reklaitis, G., and Nagy, Z. (2018). Resilience and risk analysis of fault-tolerant process control design in continuous pharmaceutical manufacturing. *Journal of Loss Prevention in the Process Industries*, 55.

[SysIdTool, 2020] SysIdTool (2020). MATLAB. https://de.mathworks.com/products/sysid.html. [Online; accessed 26-July-2020].

[xtalks, 2016] xtalks (2016). Continuous and batch pharmaceuticals manufacturing. https://xtalks.com/continuous-and-batch-manufacturing-pharmaceuticals/. [Online; accessed 26-July-2020].